YY-DM642 说明书



2011-4

目录

第一章 YY-DM642 硬件安装说明

1.1 硬件清单

1.1.1 DSP 系统板

1) YY-DM642 系统板 1 块。
2) 系统板+5V 供电电源 1 个。

1.1.2 仿真器（可选）

1) TDS510USB PLUS 仿真器 1 个(TDS560USB PLUS 仿真器 1 个)
2) USB 电缆连接线 1 根，两边插头一扁一方。

1.1.3 摄像头（可选）

1) 标准 PAL 制摄像头 1 个。
2) 摄像头供电+12V 电源转接线 1 根。 视频转接线 1 根，一端为 Q9 插头连接摄像头，另一端为插头连接 DM642 系统板。

1.1.4 TV 显示器（可选）

1) 标准 PAL/NTSC 制 TV 显示器 1 个。
2) 显示器信号转接线 1 根。

第二章 YY-DM642 评估板技术参考手册

2.1 YY-DM642 评估板简介

这一章主要给出了 YY-DM642 评估板的简要概述、软件支持、系统组成、主要功能、系统的技术指标和电路板的图示。

2.1.1 概述

YY-DSP 嵌入式图像处理模块是西安悦翼电子科技有限公司研发的一款针对各种视频应用的具有 10/100M 以太网接口的独立的模板，基于 TI 公司的多媒体处理芯片 TMS320DM642，开发环境为 CCS。用户可以对其内部的高速 DSP 系统用标准的 C 语言编程，实现自己的嵌入式应用。通过连接标准的 DSP 仿真机，用户在成熟友好的 CCS 集成开发环境下，用标准的 C 语言编写自己的嵌入式图像算法和控制算法，通过上位机的程序编辑来获取目标板的烧写文件，然后就可以脱离 PC 机单独运行。目标板加电后可独立完成多媒体处理、数据传输等功能。本系统含有板级底层驱动源码及应用层开发例程及图像处理库函数并且可以支持新的图象处理算法的开发。该系统支持图像产品开发、网络流媒体、视频会议系统、图像算法研究、目标跟踪、多路图像监控、智能交通、生物识别、医学图象处理等应用领域。

2.1.2 软件支持

1）操作系统支持 Win2000(SP2)或 WinXP

2）TI 公司的 CCS 集成开发环境编程（提供基于 DSP/BIOS 的实时操作系统软件平台，以及相应底层驱动 API）

3）视频采集驱动、显示驱动、IMGlib64x

4）视频应用演示

5）TCP/IP 协议栈，标准的 TCP/IP socket 编程，兼容所有操作系统

系统实物图 2-1 所示

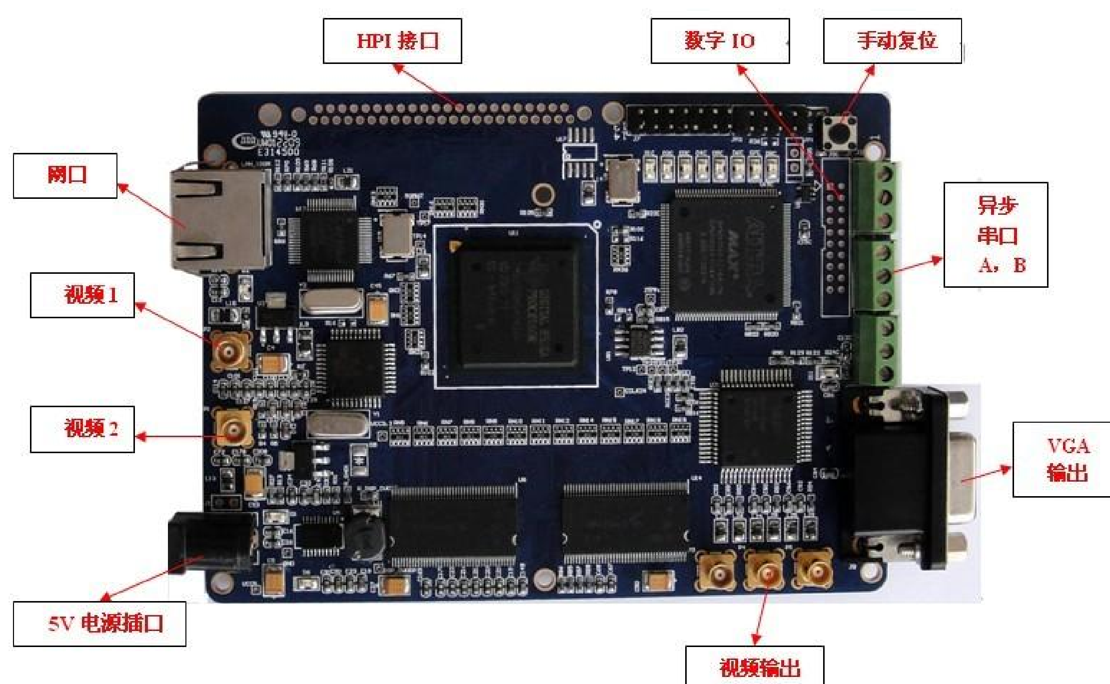


图 2-1　新版 YY-DSP 嵌入式图像处理模块

2.1.3　系统组成及主要功能

系统组成框图如图 2-2 所示。

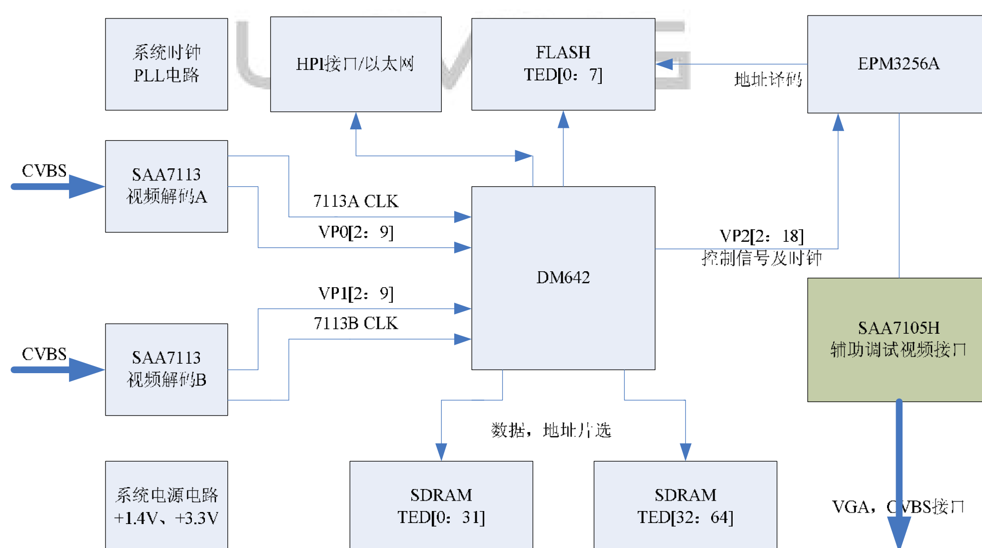


图 2-2 系统组成原理框图

2.1.4　主要功能如下

采用当前性能最强大的、专为视频应用而优化的高速数字信号处理器（DSP）TMS320DM642，主频为 600MHz；

2 路标准 PAL 或 NTSC 制模拟视频输入（CVBS 复合视频信号或 S 端子信号输入），1 路标准 PAL/NTSC 制模拟视频输出（CVBS 复合视频信号或 S 端子信号输出，VGA），用于预览；

2 路 RS232/RS422/RS485 标准软件可配置的异步串行通信口，可用于控制云台，实现镜头的推拉、水平和俯仰转动；

数字量输入 / 输出各 8 路，实现现场环境的监测和控制；

RJ45 标准以太网接口，实现视频服务器或网络摄像机功能；

2.1.5 系统技术指标

表 1 YY-DSP 模块连接器

主处理器：	TMS320DM642 ，工作主频高达 600MHz ，处理能力可达 5760MIPS
SDRAM：	64M，工作时钟 100MHz ，256 kbit I2C EEPROM
Flash：	8M 位，70ns（20 年数据保存，1,000,000 次擦写）
视频输入：	2 通道， PAL/NTSC 标准模拟视频信号（CVBS 或 Y/C 可选），RCA 及 S 端子接口，最大输入范围：0～1 VPP
视频输出：	1 通道，PAL/NTSC 标准模拟视频信号（CVBS＋Y/C 可选，VGA），最大输出范围：CVBS：0～1.23 VPP，Y：0～1VPP，C：0～0.89 VPP NTSC：720×525@30 帧 / 秒 PAL：720×625@25 帧 / 秒
异步串口：	2 通道， RS232/RS422/RS485 可编程配置接口 传输率：RS232：1Mbaud；RS422/RS485：1.92Mbaud
数字 I/O 及 LED：	8 路开入、8 路开出，接口输出 开入：VIH：2.0V～5.5V，VIL：-0.5V～0.8V 开出：VOH：2.4V@-4mA，VOL：0.5V@8mA 8 个用户自定义的 LEDs
外部触发输入	16 个 DSP 预留 IO
外部复位	1 个 TTL 输入的外部强制硬件复位控制信号（低电平复位）
以太网接	10M/100Mbase-TX 标准，标准的带绿、黄 2 个 LED 指示灯的

口：	RJ45 连接器，绿灯指示连接状态，黄灯指示数据传输或传输速度
仿真接口	内部 DSP 的 JTAG 信号已经连接到外部端口，不需打开外壳即可连接 DSP 仿真机在 CCS 软件环境下实时仿真调试
工作温度：	-20～60℃ -40～85℃
机械尺寸：	（120mm×85mm）

2.2 YY-DSP 模块接口

2.2.1 概述

YY-DSP 嵌入式图像处理模块拥有网络接口、串口、GPIO 接口与外界通信，用户可以根据自己的使用场合选择合适的通信方式。

YY-DSP 模块支持标准的 TCP/IP 协议，用户可以利用网络和外界通信，速度为 10M/100M 速度，网络可以用于大量数据传输的场合，比如视频压缩码流的传输。

YY-DSP 模块拥有两个串口，支持 RS232/RS485 协议，串口可以用于云台控制等场合。

YY-DSP 模块拥有 16 个 GPIO 接口，8 个输入，8 个输出，GPIO 可以用于简单的控制信号输出、键盘译码等。

YY-DSP 模块拥有 8 个用户自定义 LED，用户可以根据自己需要设置这些 LED 指示含义。

2.2.2 网络接口使用

在独立工作方式下，YY-DSP 模块的以太网接口自动使能，用 LXT971 物理层芯片扩展了一个 10/100Mbit 的以太网接口，连接器为 RJ-45 标准以太网连接器。板子上有两个指示灯指示网络连接状态和数据传输。

为了加速其高档 DSP 的网络化进程，TI 公司结合其 C6000 系列芯片推出了 TCP/IP NDK (Network Develop’s Kit)开发套件。NDK 设计的目的是要提供一个完整的 TCP/IP 功能环境。NDK 通过编程接口与本地操作系统 DSP/BIOS 和底层硬件相互隔离，本地操作系统 DSP/BIOS 被抽象成一个操作系统适应层(OS Adaptation Layer)，底层硬件被抽象成一个硬件抽象层(Hardware Abstraction

Layer)，两个抽象层的函数库分别为 OS.LIB 和 HAL.LIB ,它们是 NDK 与本地操作系统和底层硬件的接口。NDK 主要的组件包括：

1) 支持 TCP/IP 协议栈程序库。其中主要包含的库有：支持 TCP/IP 网络工具的库，支持 TCP/IP 协议栈与 DSP/BIOS 平台的库，网络控制以及线程调度的库(包括协议栈的初始化以及网络相关任务的调度)；

2）示范程序。其中主要包括 DHCP/Telnet 客户端，HTTP 数据服务器示范等，这些示范程序为用户学习利用 NDK 提供了参考。

NDK 采用紧凑的设计方法，实现了用较少的资源耗费来支持 TCP/IP。从实用效果看，NDK 仅用 200~250K 程序空间和 95K 数据空间即可支持常规的 TCP/IP 服务，包括应用层的 Telnet、DHCP、HTTP 等。为了最大限度地减少资源消耗， TI 为其 NDK 采用了许多特殊的技巧，重要的有：

1) UDP Socket 和 RAW Socket 不使用发送或接收缓冲区；

2) TCP Socket 使用发送缓冲区，接收缓冲区依配置文件而定；

3) 低层驱动程序与协议栈之间通过指针传递数据，不对包进行复制拷贝；

4) 设置专门的线程清除存储器中的碎片和检查存储器泄露。

因此，NDK 很适合目前嵌入式系统的硬件环境，是实现 DSP 联网通信的重要支撑工具。NDK 的软件开发环境是 TI 的开发工具 CCS (Code Composer Studio)。它包含有实时操作系统 DSP/BIOS 和主机与目标板之间的实时数据交换软件 RTDX。

DM642 上网络程序的开发，首先要正确配置系统网络工作方式，具体可参见相关文档。当配置完系统后，就可以像在 PC 下一样利用标准 SOCKET 函数操作网口。DM642 上的网络编程用户首先应该熟悉网络底层配置的相关知识，关于这些内容在 TI 的文档中有详细介绍。

2.2.3 串口使用（可选功能）

YY-DSP 模块有两个 RS232 串口。TL16C752B 是一个带有 64 字节 FIFO 的通用异步收发器（UART），自动硬件/软件流控制，数据率达到 3Mbps。MAX3243 是 RS232 串口采用的电平转换芯片。

串口通信波特率可以用户自己设置，见相关程序。

EVMDM642_UART_Handle hUart;

```
EVMDM642_UART_Config uartcfg =
{
        0x00,  // IER
        0x57,  // FCR - FIFO Mode, 16 character trigger level
        0x03,  // LCR - 8 bits, no parity, 1 stop
        0x00  // MCR
    };
// Open UART
//设置波特率 9600kbps
hUart = EVMDM642_UART_open(EVMDM642_UARTA, EVMDM642_UART_BAUD9600,
&uartcfg);
```

通过串口发送数据函数

void EVMDM642_UART_putChar(EVMDM642_UART_Handle hUart, Uint16 data)

通过串口接收数据函数

Int16 EVMDM642_UART_getChar(EVMDM642_UART_Handle hUart)

关于串口使用见程序 UARTControlA，该程序把串口 A 设置为 RS232 工作方式，通过串口控制云台转动。

2.2.4 GPIO 使用（可选功能）

YY-DSP 模块 GPIO 接口通过 CPLD 扩展，GPIO 地址映射在 CPLD 中，输入寄存器地址 0x90080011，输出寄存器地址 0x90080012。GERNERAL_IO0-20 的管脚顺序从 1 脚开始呈左右排列。GERNERAL_IO19-20 为 GND，VCC3.3。

```
#define EVMDM642_GPIOIN        0x11     GERNERAL_IO2—IO9
#define EVMDM642_GPIOOUT       0x12     GERNERAL_IO10—IO17
```

2.2.5 LED 指示灯使用（可选功能）

DM642 视频处理板上 LED 指示灯与 CPLD 连接，地址映射在 CPLD 中，LED 寄存器地址 0x90080013。

```
#define EVMDM642_LED          0x13
```

2.3 接口说明

表 2　YY-DSP 模块连接器

连接器	管脚数	功能
J5C	20	用户自定义 GPIO
J6C	44	HPI 接口
J7	14	DM642 JTAG
J8	15	VGA 接口
J9	8	RJ45 Internet
P3	2	SAA7105 CVBS 输出(VP2)
P4	2	SAA7105 CVBS 输出
P5	2	SAA7105 CVBS 输出
JP1	2	备用手动复位跳线
JP3	10	CPLD JTAG
P1	2	CVBS 输入接口 1(VP0)
P2	2	CVBS 输入接口 2(VP1)
J2	3	A_TXD/A_RXD/A_RTS
J4	3	A_CTS/B_TXD/B_RXD
J5	3	B_RTS/B_CTS/GND
JCK1	3	＋5V 电源接口

第三章 YY-DM642 实验系统的安装说明

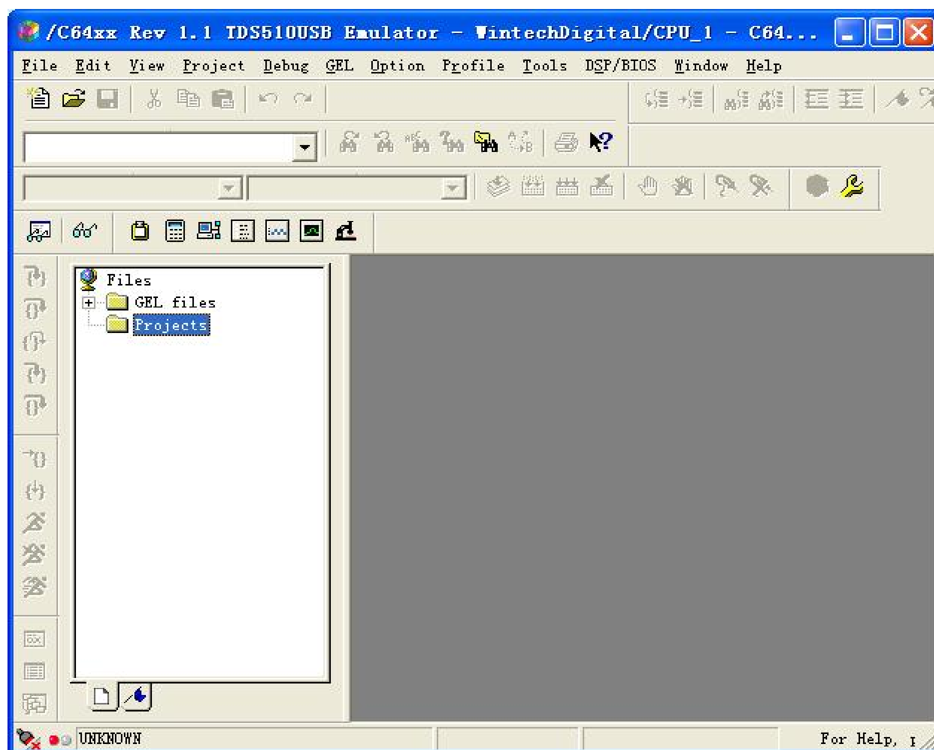
3.1 软件安装

3.1.1 环境搭建：

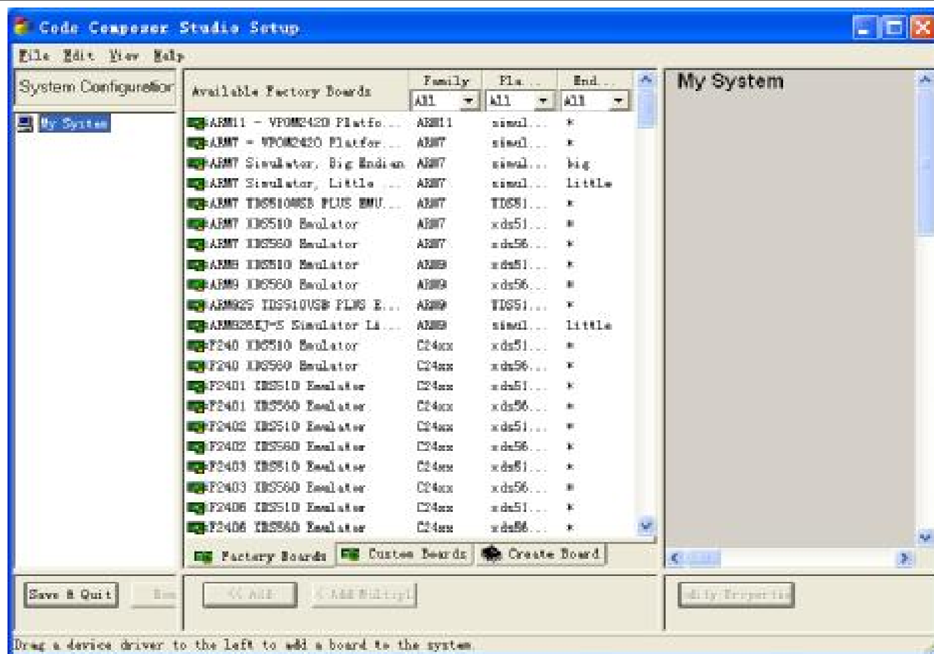
1、 安装 ccs3.1（本例路径为：C:\CCStudio_v3.1），默认配置，只需点击下一步即可完成安装。

2、 配置仿真器（以 TDS510USB Pluse 仿真器为例），首先安装仿真器驱动（仿真器自带的光盘上），接下来配置 ccs

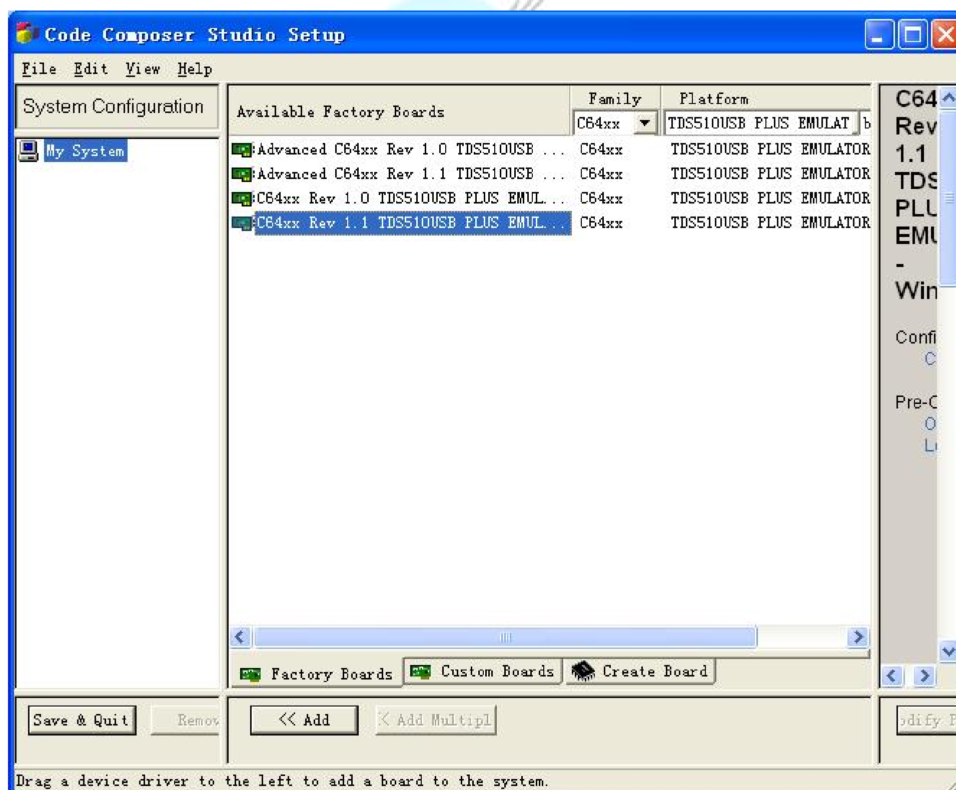
（1）、启动 ccs，开始 **à** 所有程序 **à** Texas Instruments **à** Code Composer Studio 3.1 **à** Code Composer Studio 3.1，显示如下图：



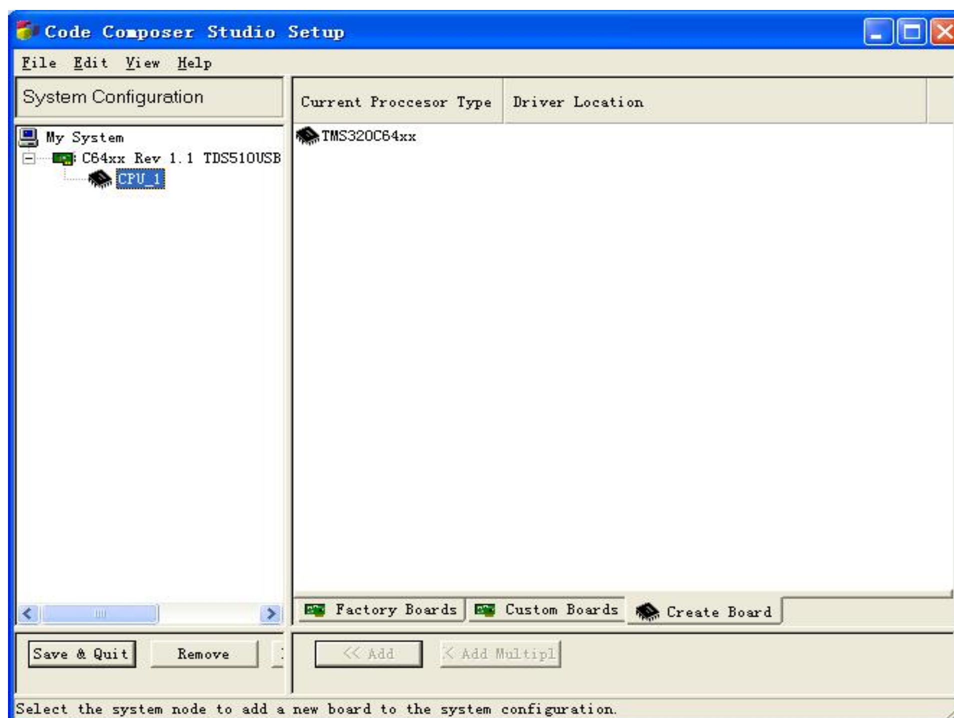
（2）、点击上图菜单栏中的 File **à** Launch Setup，结果如下图：



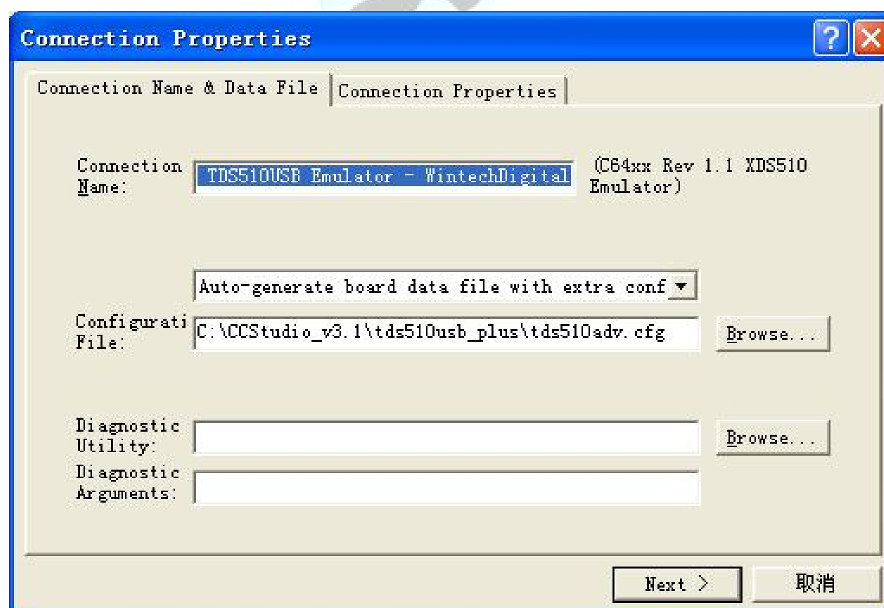
（3）、在上图的 Family、Plateform 中分别选择 C64xx 和 TDS510USB PLUS EMULATOR，如下图所示：



（4）、选择上图中过滤出的 C64xx Rev 1.1 TDS510USB PLUS EMULATOR，点击 << Add 其结果如下图所示：

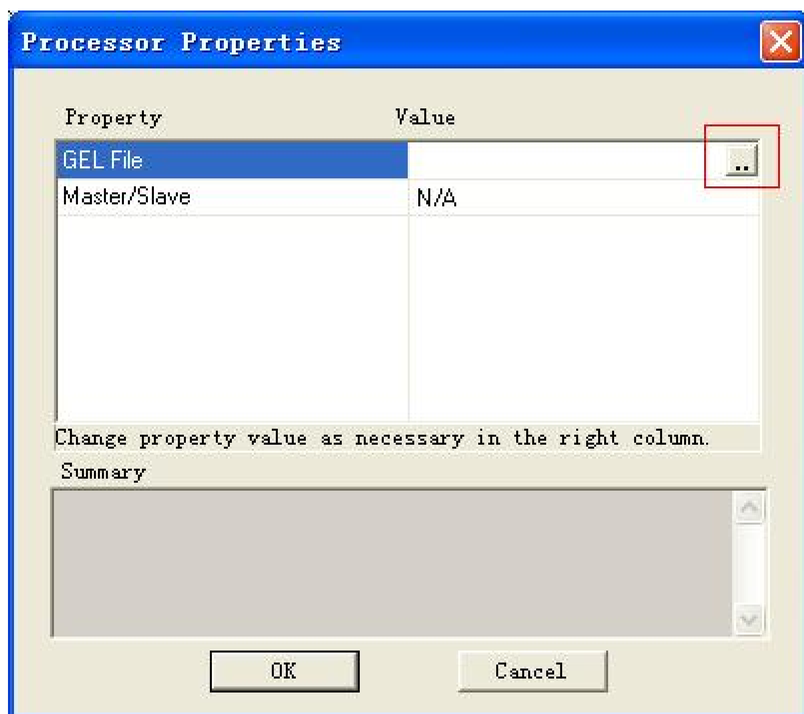


（5）、在上图中的左侧栏 C64xx Rev 1.1 TDS510USB 上右键，选择 Properties，配置如下：

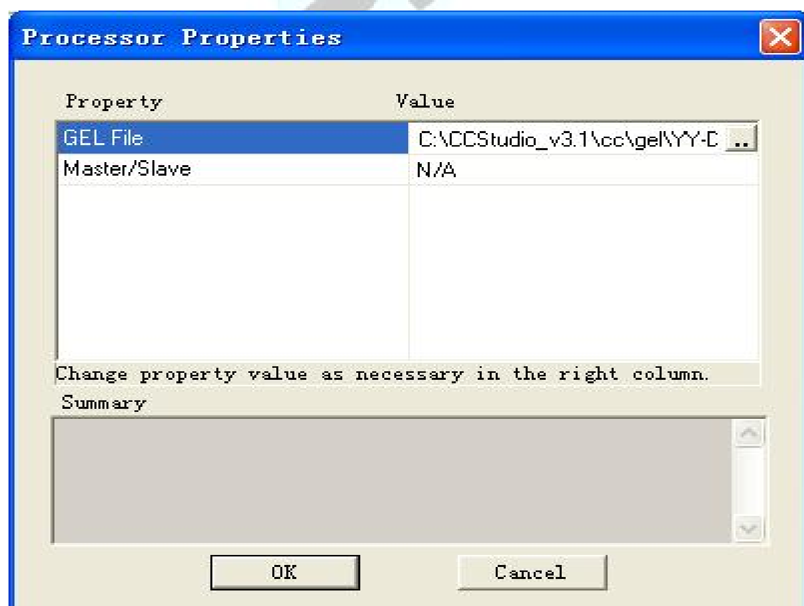


Next 继续，直至 finish。

（6）、配置 gel 文件：在（4）图中的 CPU_1 上右键，选择 Properties，出现下图：

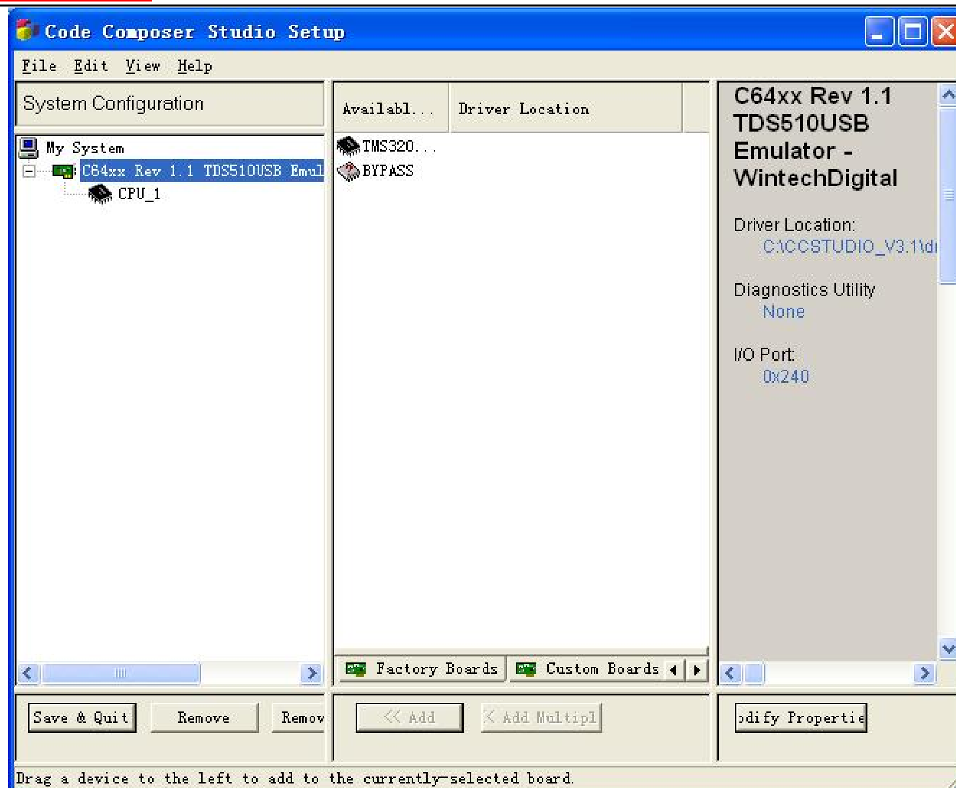


（7）、将光盘中的 YY-DM642.gel 拷贝至 CCStudio_v3.1\cc\gel\下，点击上图红框内按钮添加 YY-DM642.gel 所在路径。如下图：



点击 OK，完成 gel 文件配置。

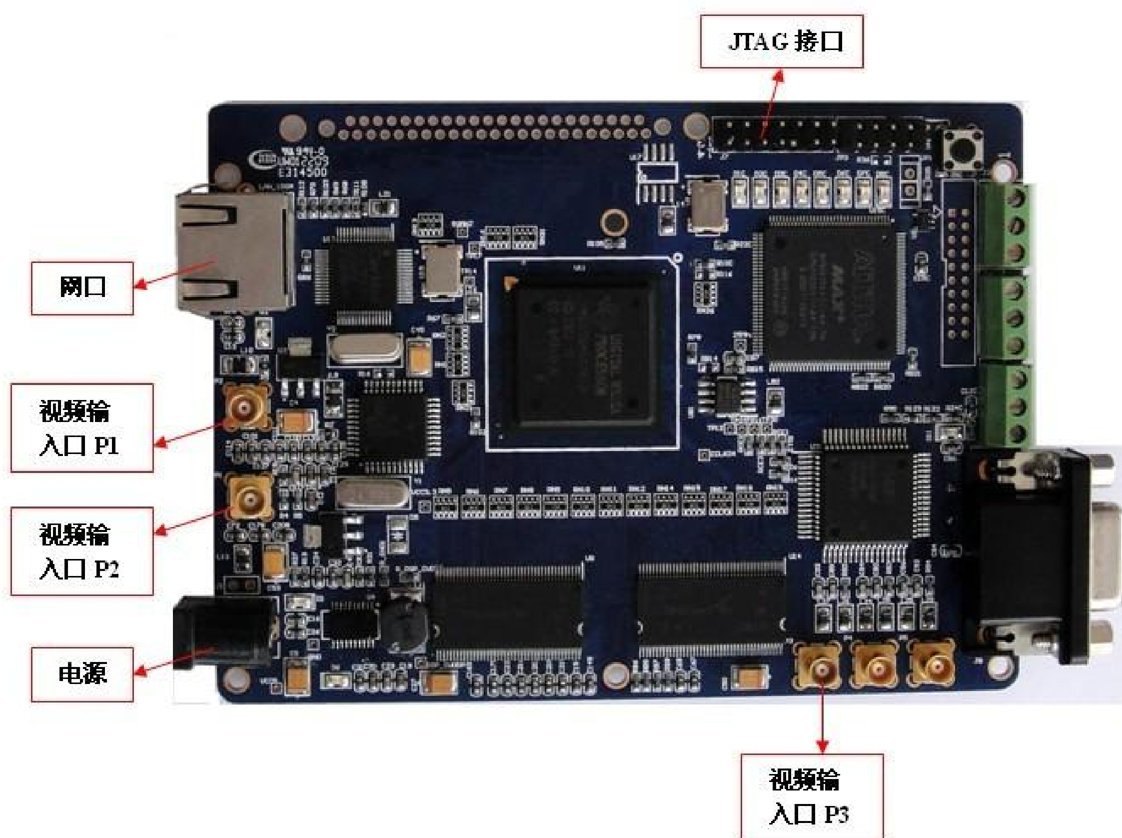
（8）、完成以上步骤，又回到此界面：



点击 Save & Quit 完成仿真器配置。此时弹出消息框，点击是，重启 ccs。

3、 连接仿真器

（1） 将仿真器 jtag 口连接到板子的 J7 上，网线（交叉网线）连接板子和 pc，视频输入连接到 P1，将视频输出连接到板子的 P3。硬件连接图如下：



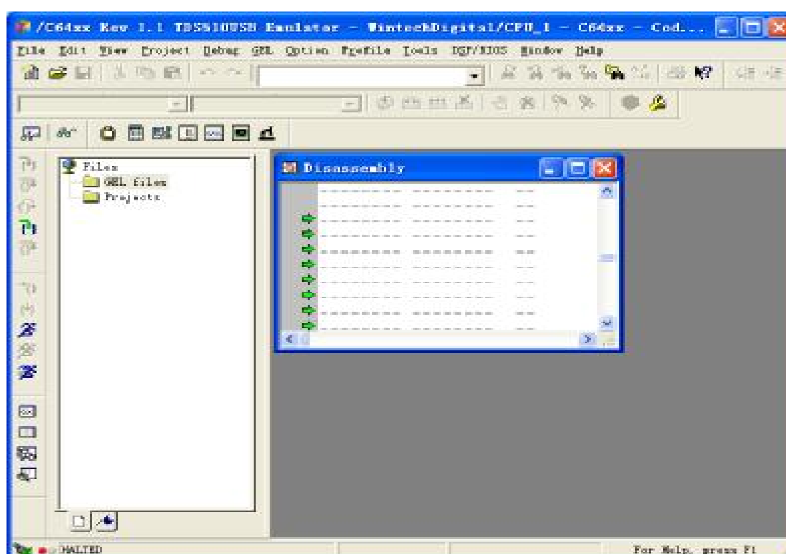
打开监视器，将仿真器的 USB 连接到电脑上，这时会提示找到新硬件，选择如下：



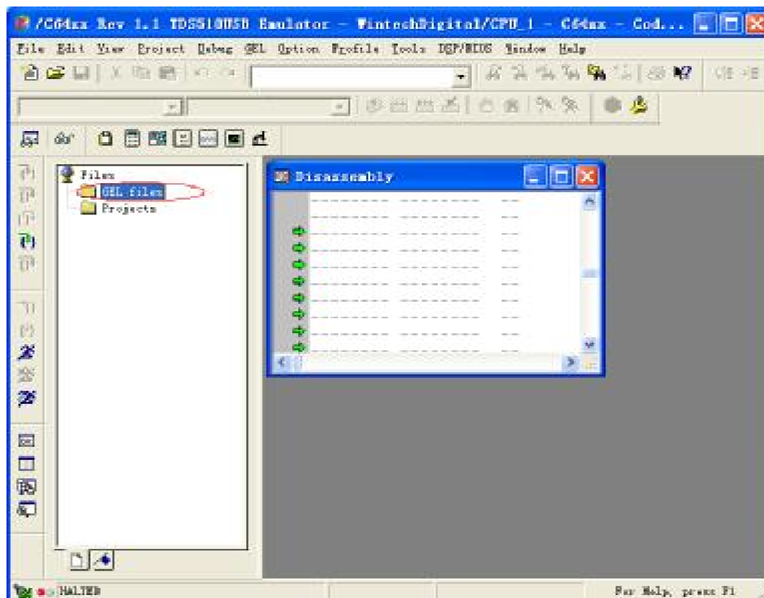


点击下一步，直至安装完成。

（2）板子上电（5V/2A），这时监视器会输出 colorbar，打开 ccs 窗口，点击菜单栏中 Debug à Connect，其结果如下图：表示已连接成功



4、 加载 gel 文件，右键下图红圈，选择 Load GEL，然后找到 YY-DM642.gel 所在位置。



运行程序（直通程序，不带网络功能）：将光盘中 YY-DM642 程序\loopback 下的 VPOLoopback 文件夹拷贝至 C:\CCStudio_v3.1\MyProjects，可运行的程序为：C:\CCStudio_v3.1\MyProjects\VPOLoopback\Debug\LYVideoprocess.out

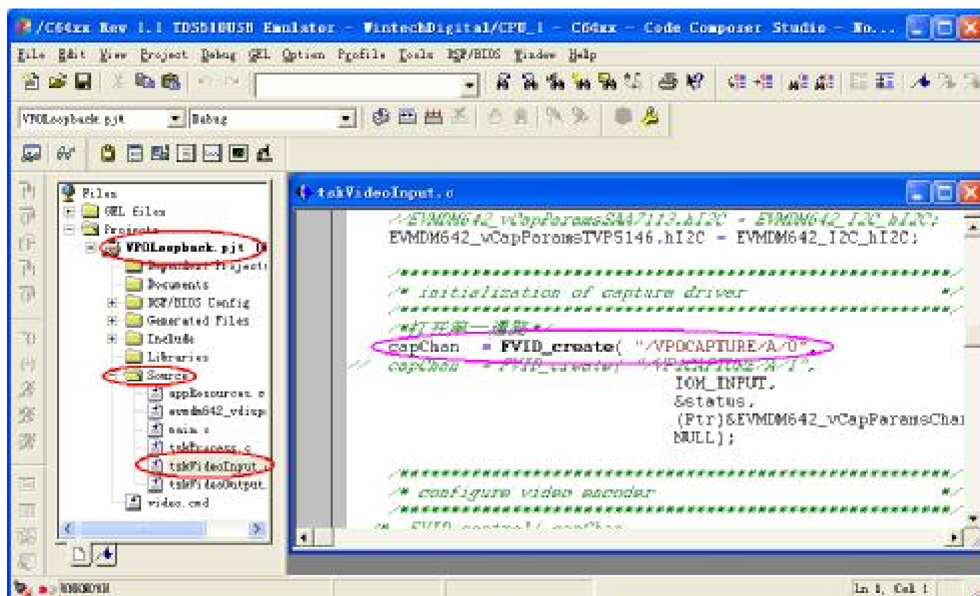
注：程序路径不可含有中文！

3.2 程序运行演示说明

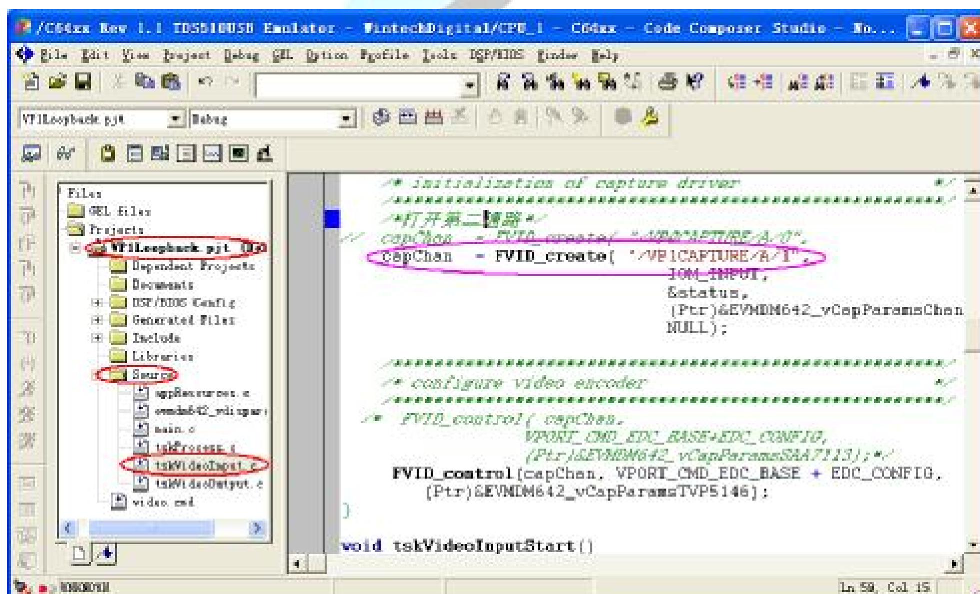
以下所运行的程序均在光盘下 YY-DM642 程序文件夹下；

所运行的程序是以 P1 口为输入还是以 P2 口为输入，通过以下方式来查看：

打开所要运行的程序工程，以 VPOLoopback 的接口为 P1 例：打开工程并找到红框所示，打开 tsKVideoInput.c，如粉框所示内容，即表示输入接口为 P1 口。



打开所要运行的程序工程，以 VP1Loopback 的接口为 P2 例：打开工程并找到红框所示，打开 tsKVideoInput.c，如粉框所示内容，即表示输入接口为 P2 口。



3.2.1 loopback 程序

3.2.1.1 VP0Loopback（模式为：输入 à 输出）

实验步骤：

1．将视频输入线接入 P1 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，;

3．接着在菜单栏中单击 File à Load Program，找到 VP0Loopback\Debug 下的*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project－》Build（）；停止程序运行，点击。

此时有程序正在运行，而我们想运行另一程序，具体操作如下：先点击，停止正在运行的程序，然后菜单栏 Debug à Reset CPU（如果不执行此操作，510 仿真器下一次下载程序速度将会很慢），继续菜单栏 File à Load Program 下载想要运行的程序即可。

3.2.2.1　VP1Loopback（模式为：输入 à 输出）

实验步骤：

1．将视频输入线接入 P2 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，;

3．接着在菜单栏中单击 File à Load Program，找到 VP1Loopback\Debug 下的*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project－》Build（）；停止程序运行，点击。

3.2.2.3　H.263_loopback（模式为：输入 à H.263 编码 à 输出）

实验步骤：

1．将视频输入线接入 P1 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，;

3．接着在菜单栏中单击 File à Load Program，找到 H.263_loopback\bin 下的*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project—》Build（）；停止程序运行，点击。

3.2.2.4 LYVP0H264Loopback（模式为：输入 à H.264 编码 à 输出）

实验步骤：

1．将视频输入线接入 P1 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，；

3．接着在菜单栏中单击 File à Load Program，找到 LYVP0H264Loopback\Debug 下的*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project—》Build （）；停止程序运行，点击。

3.2.2.5 LYVP1H264Loopback（模式为：输入 à H.264 编码 à 输出）

实验步骤：

1．将视频输入线接入 P2 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，；

3．接着在菜单栏中单击 File à Load Program，找到 LYVP1H264Loopback\Debug 下的*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project—》Build （）；停止程序运行，点击。

3.2.2 jpeg 网络摄像机程序：

说明：JAVA 插件在光盘 JAVA applet 文件夹下

3.2.2.1 jpeg_motion

实验步骤：

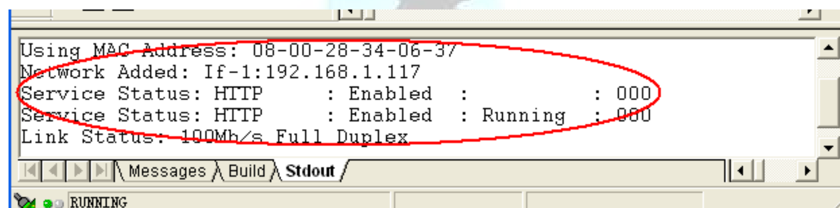
1.将视频输入线接入 P2 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2.给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

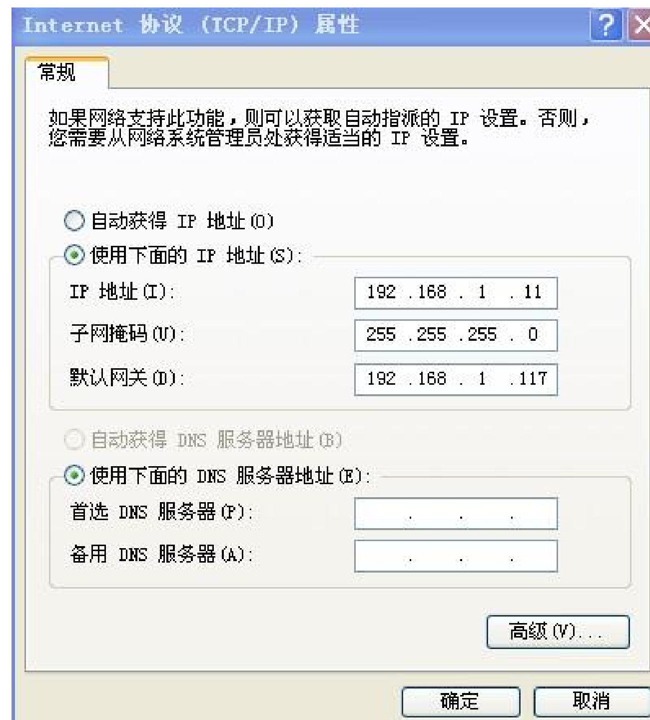
单击菜单栏中 Debug à Connect，；

3.接着在菜单栏中单击 File à Load Program，找到 jpeg_motion\bin\Debug 下的 *.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project－》Build （）；停止程序运行，点击。

4.运行成功如下图所示



在 ccs 窗口中看到板子的 ip 地址为 192.168.1.117，更改主机 ip 使之与板子的 ip 在同一段，设置可以如下图所示：



打开 IE 浏览器，在地址栏输入 192.168.1.117 即可看到视频图像（需要安装 JAVA 插件）。通过监视器也可看到输出图像。下图为通过网络显示的视频截图：



3.2.2.2 jpeg_netcam

（通过 IE 浏览器两通道可选，可选择图像质量）

实验步骤：

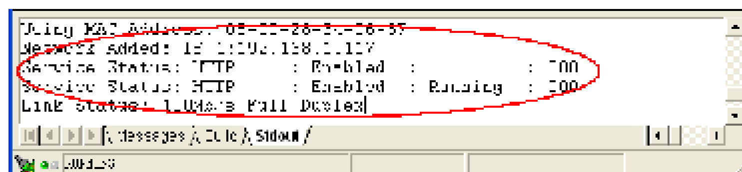
1.将视频输入线接入 P2 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

2.给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

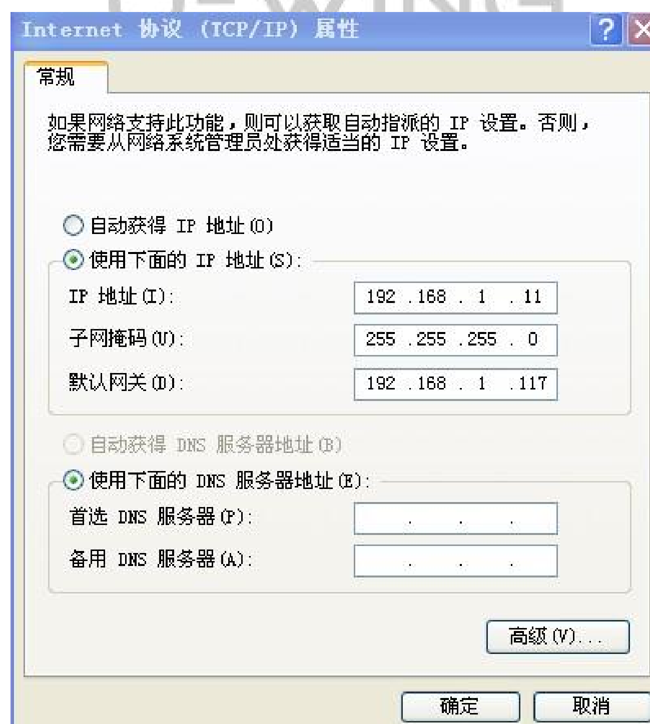
单击菜单栏中 Debug à Connect，；

3.接着在菜单栏中单击 File à Load Program，找到 jpeg_netcam\bin\Debug 下的 *.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project－》Build （）；停止程序运行，点击。

4.运行成功如下图所示



看到板子的 ip 地址为 192.168.1.117，更改主机 ip 使之与板子的 ip 在同一段，设置可以如下图所示：



打开 IE 浏览器，在地址栏输入 192.168.1.117 即可看到视频图像（需要安装 JAVA

插件）。下图为通过网络显示的视频截图：



3.2.2.3　jpeg_network

实验步骤：

1.将视频输入线接入 P2 口，视频输出线接入 P3 口，将仿真器 JTAG 接入 J7，将仿真器 USB 端连入电脑；

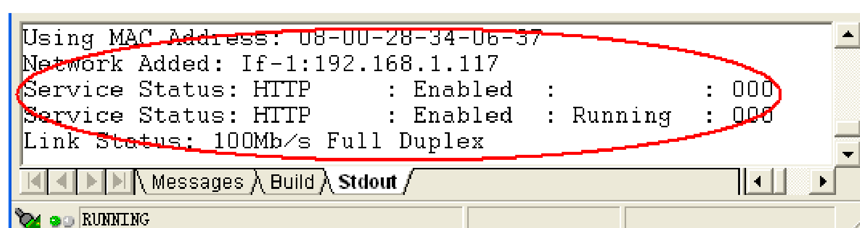
2.给仿真器和 YY-DM642 上电（5V/2A），打开 CCS3.1，

单击菜单栏中 Debug à Connect，；

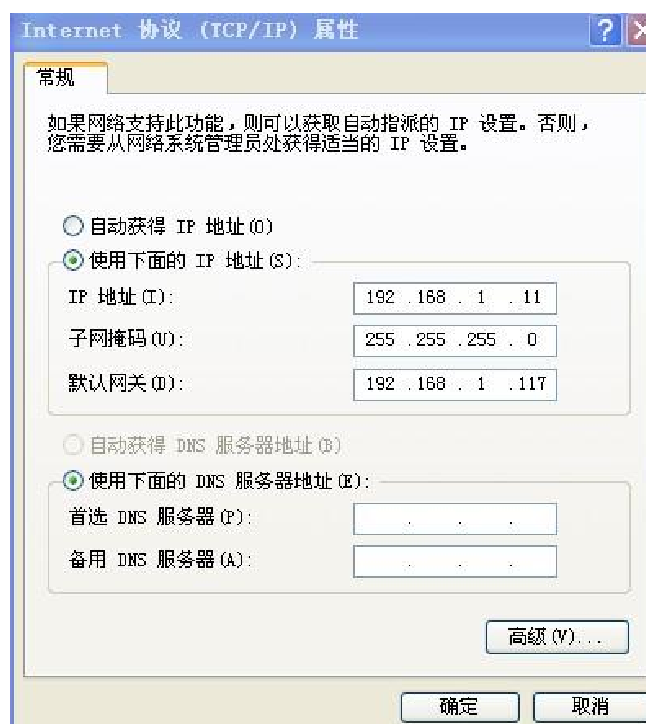
3.接着在菜单栏中单击 File à Load Program，找到 jpeg_network\bin\Debug 下的

*.out 文件并打开，将其下载到板子上，然后点击 Debug à Run 或运行程序，这时在监视器端可看到输出图像。如果没有 out 文件就先编译一下 Project－》Build （）；停止程序运行，点击。

4.运行成功如下图所示



看到板子的 ip 地址为 192.168.1.117，更改主机 ip 使之与板子的 ip 在同一段，设置可以如下图所示：



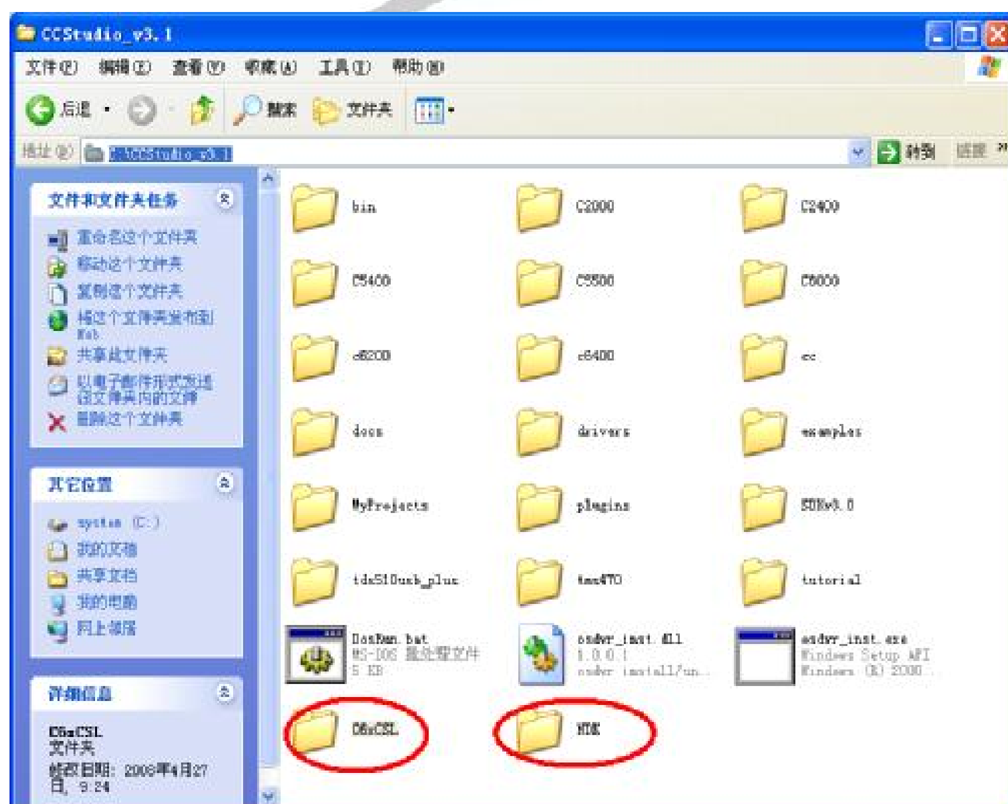
打开 IE 浏览器，在地址栏输入 192.168.1.117 即可看到视频图像（需要安装 JAVA 插件）。通过监视器也可看到输出图像。下图为通过网络显示的视频截图：



3.2.3 H.264 网络摄像机

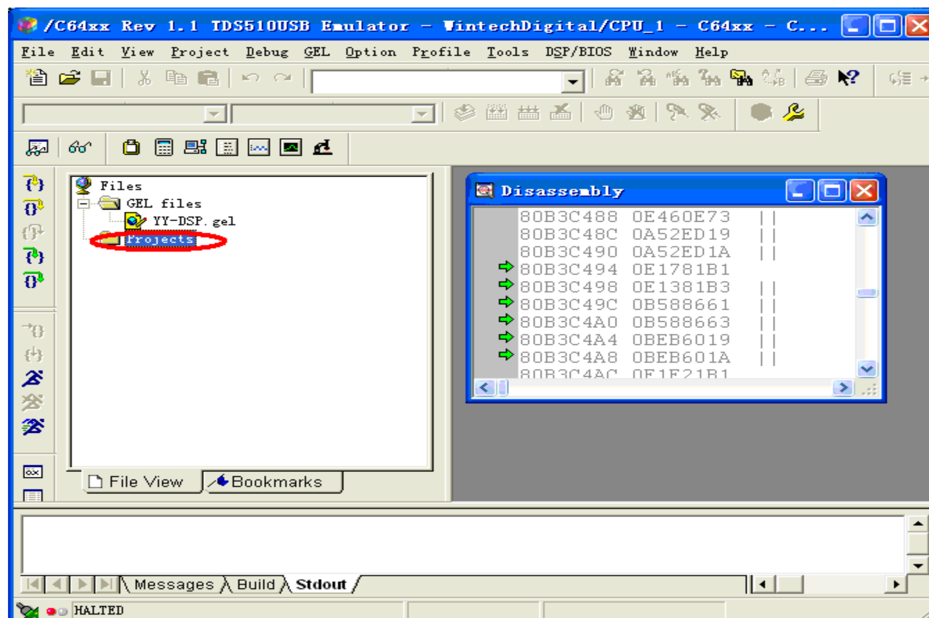
实验步骤：

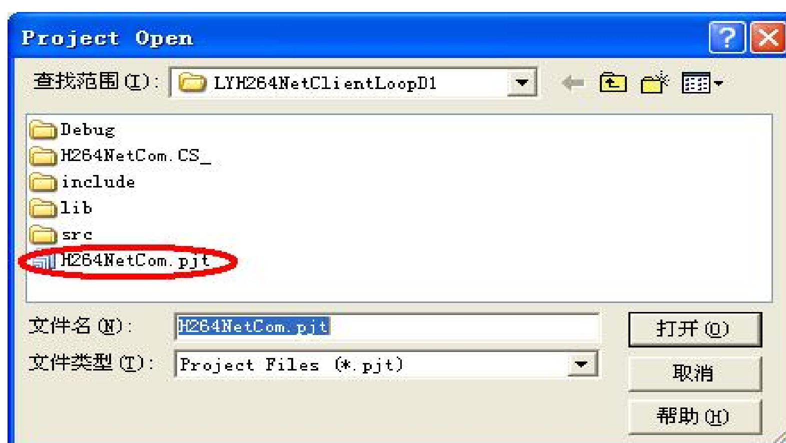
1.将光盘下的 YY-DM642 程序中的 C6xCSL.rar 和 NDK.rar 拷贝至 C:\CCStudio_v3.1 并解压至当前文件夹，其结果如下图：



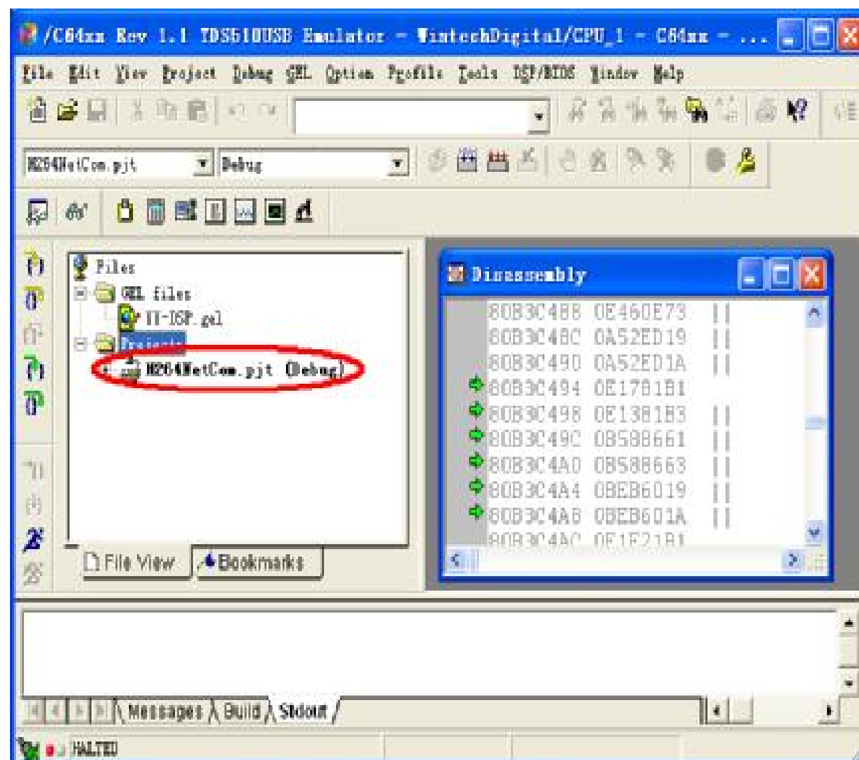
2.在 ccs 中添加工程 H264NetCom.pjt，如下图：右击红框所示选择 Open

Project 找到 H264NetCom.pjt 并打开。

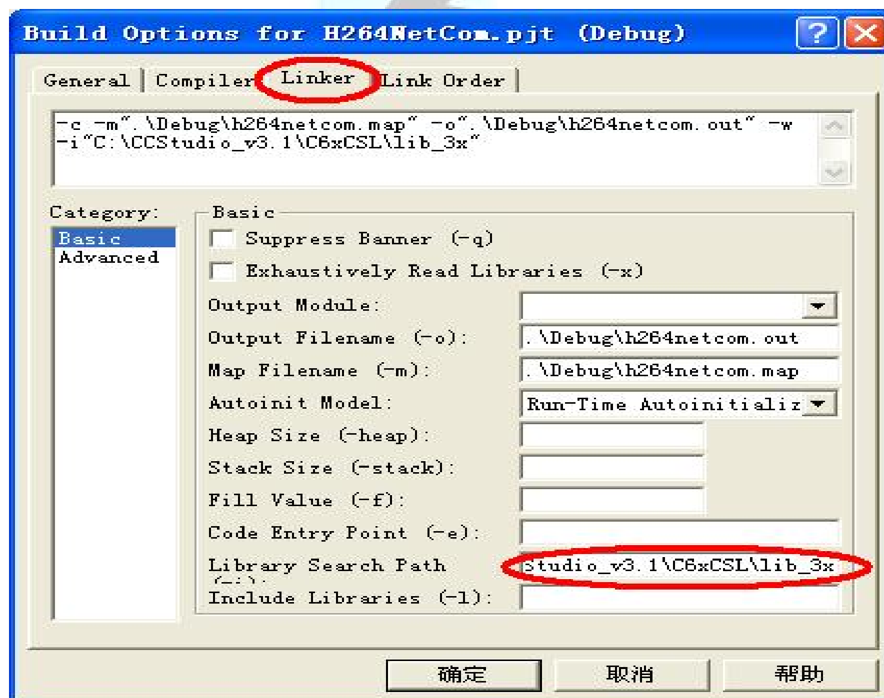


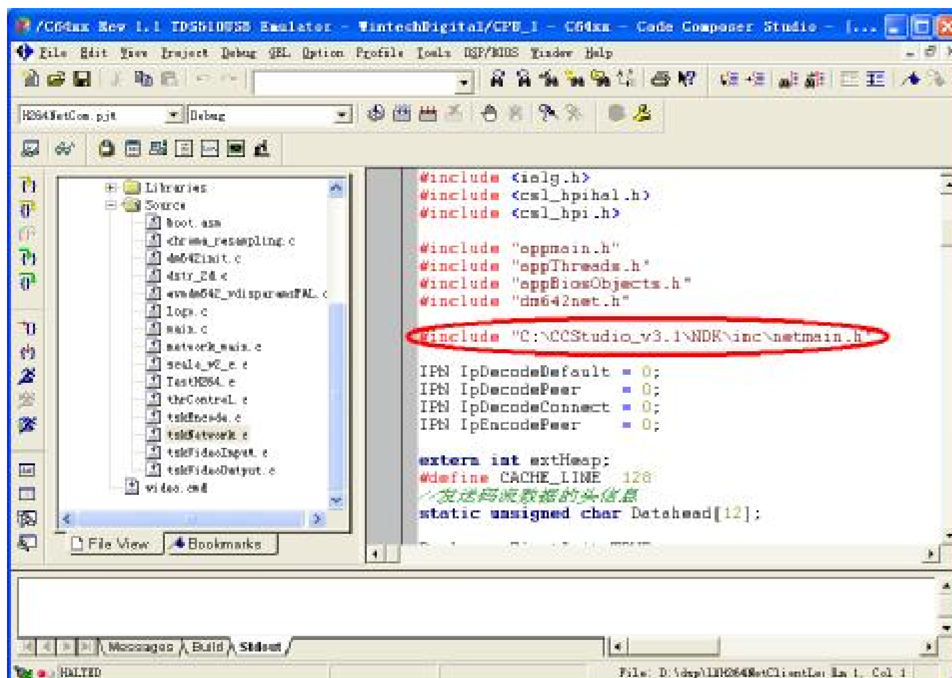


打开现象如下图：



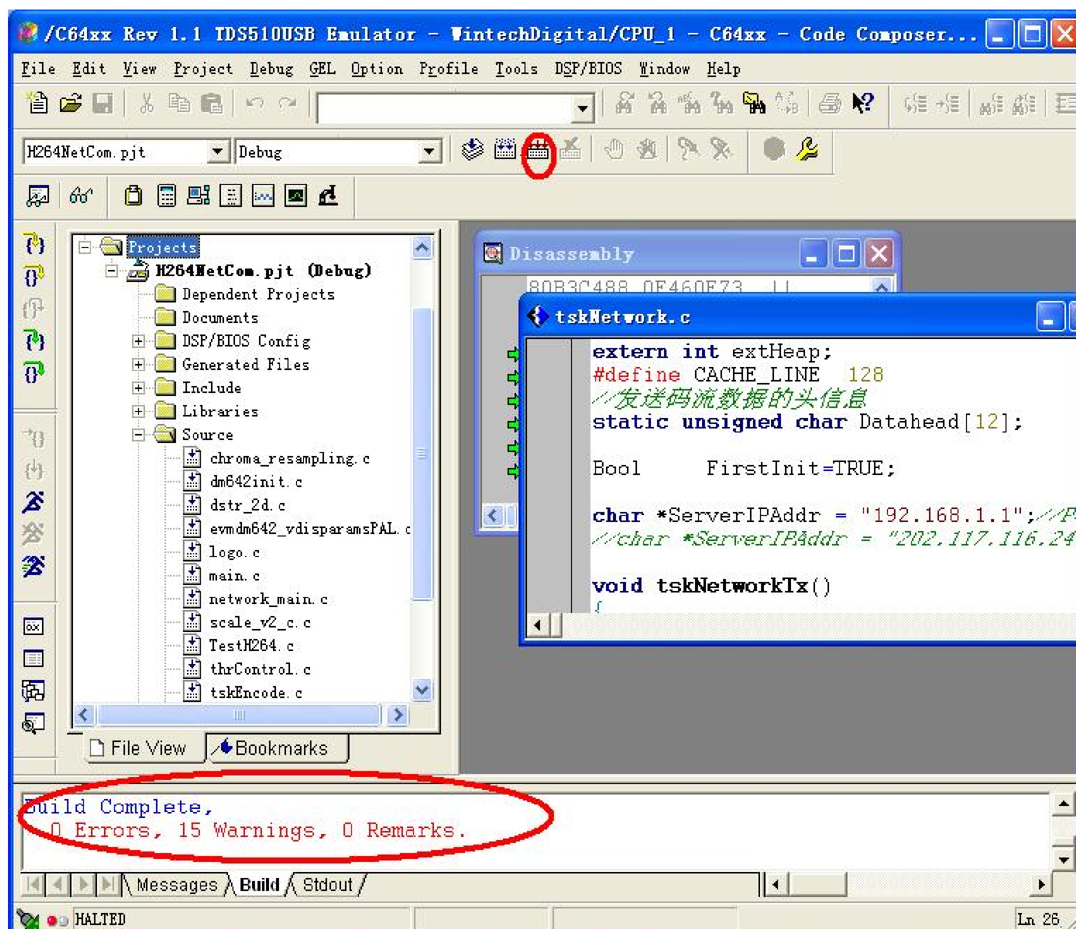
3.右击打开的工程（上图红框所示），选择 Build Options，弹出下图：





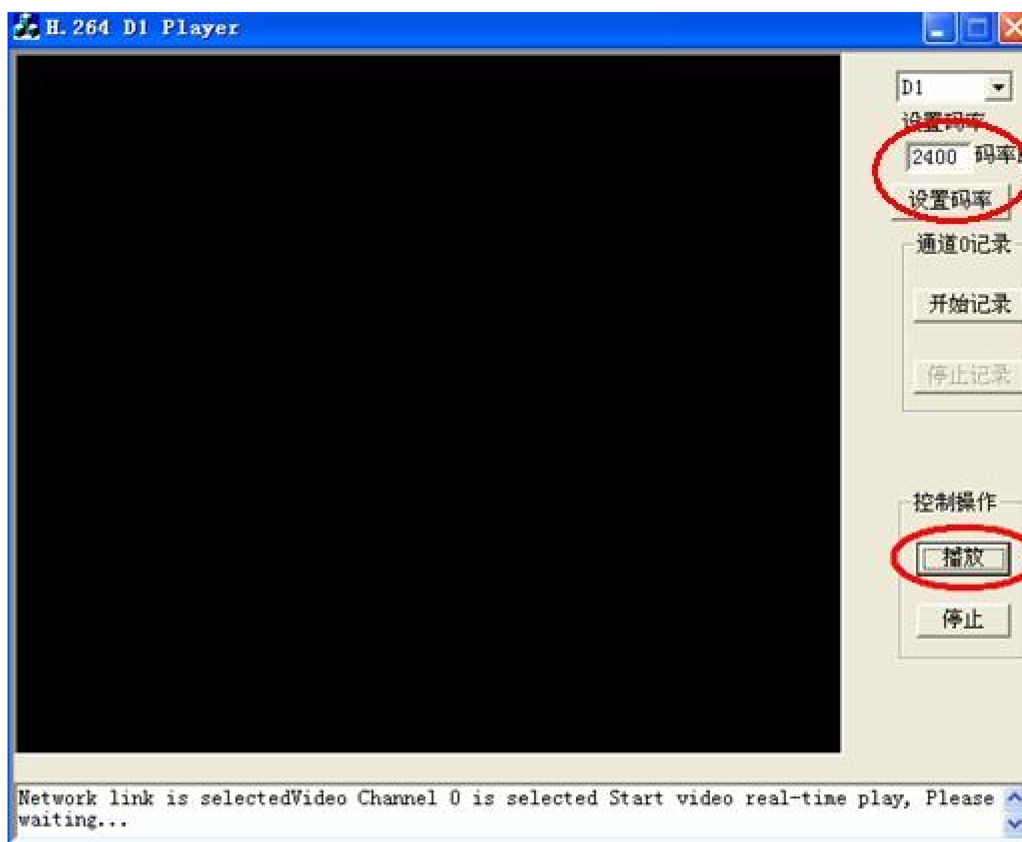
说明：以上两图所示，标明了进行 C6xCSL.rar 和 NDK.rar 的原因。一定要注意路径的一致性，例如 CCStudio_v3.1 如果安装在 D 盘目录下，则要修改红框的 C 盘为 D 盘)

4.进行编译操作，点击下图小红框所示：大红框所示信息表示编译成功。



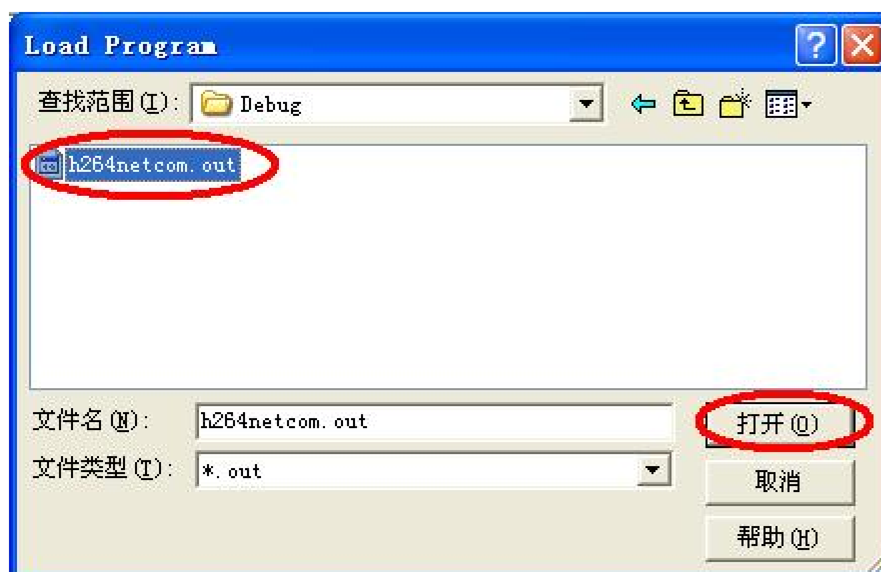
5.安装 vc6.0，才能在 pc 机端开启解码播放器。流程为：先运行播放器，点击下图所示播放按钮，修改 pc 主机的 ip 地址为程序所写服务器 ip，（将主机 ip 改为 192.168.1.1），下载并运行程序，程序运行后点击播放器界面上的设置码率，这时播放器即可显示图像。

将光盘下 YY-DM642 程序下的 LYH264NetClientLoopD1 和 LY264NetPlayer 文件夹拷贝至 C:\CCStudio_v3.1\MyProjects，并进入 LY264NetPlayer\Debug 运行 H.264 D1 Player.exe， 其界面如下图所示：

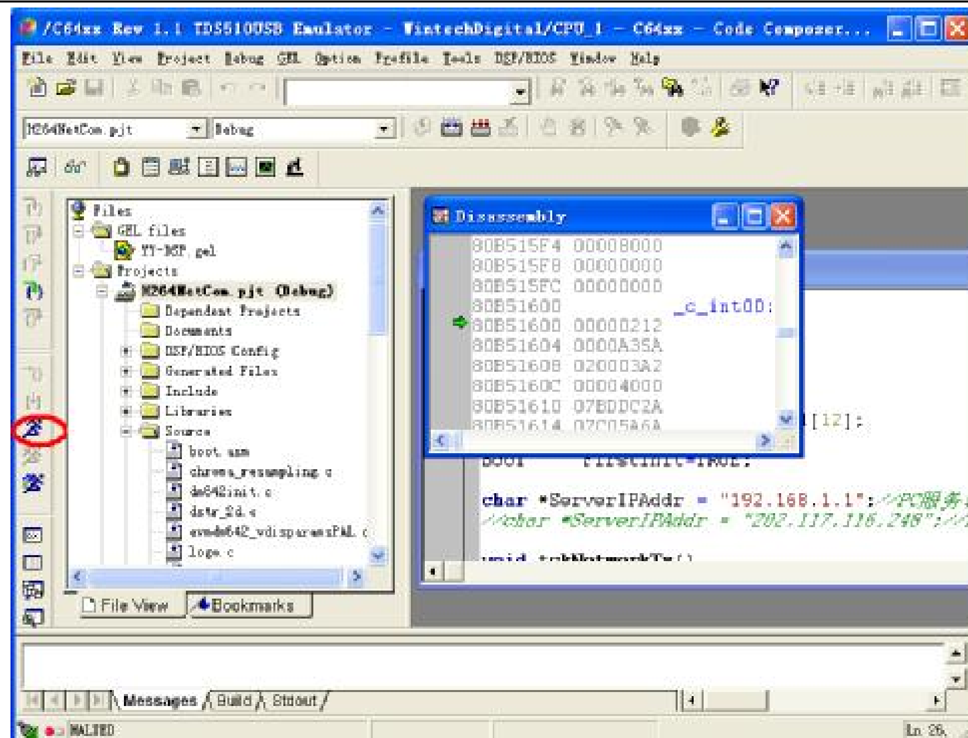


注意在点击 Debug à Run 或 运行程序之前一定要先播放 H.264 D1 Player.exe，另外还要注意码率的设置（此处设为 2400 码率）

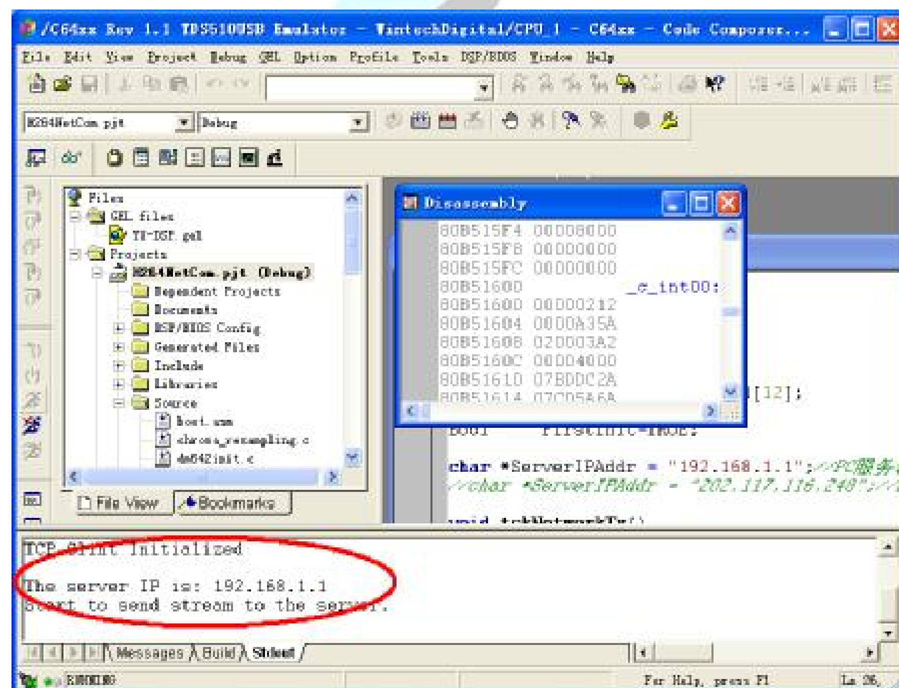
6.接下来加载程序 File à Load Program，弹出下面界面：找到 LYH264NetClientLoopD1\Debug\h264netcom.out 并打开，如下图：



等待下载完成，在 ccs 点击下图红框所示：运行程序


/C64xx Rev 1.1 TDS510USB Emulator - WintechDigital/CPU_1 - C64xx - Code Composer...
File Edit View Project Debug GEL Option Profile Tools DSP/BIOS Window Help
H264NetCom.pjt
Debug
Files
GEL files
YY-DSP.gel
Projects
H264NetCom.pjt (Debug)
Dependent Projects
Documents
DSP/BIOS Config
Generated Files
Include
Libraries
Source
boot.asm
chroma_resampling.c
dm642init.c
File View
Bookmarks
Disassembly
80B515F4 00008000
80B515F8 00000000
80B515FC 00000000
80B51600 _c_int00:
80B51600 00000212
80B51604 0000A35A
80B51608 020003A2
80B5160C 00004000
80B51610 07BDDC2A
char *ServerIPAddr = "192.168.1.1";
Messages
Build
Stdout
HALTED

其结果如下：


/C64xx Rev 1.1 TDS510USB Emulator - WintechDigital/CPU_1 - C64xx - Code Composer...
File Edit View Project Debug GEL Option Profile Tools DSP/BIOS Window Help
H264NetCom.pjt
Debug
Disassembly
char *ServerIPAddr = "192.168.1.1";
TCP Client Initialized
The server IP is: 192.168.1.1
Start to send stream to the server.
Messages
Build
Stdout
RUNNING
For Help, press F1

3.3 DSP 集成开发环境 ccs3.1 下烧写程序

烧写程序目的是为了脱机运行，这个过程是把可执行文件以可烧写文件的形式放到 Flash 中。首先需要做好准备工作。

第一步，安装 CCStudio_v2.2

第二步，安装 CCS2.2 升级版插件

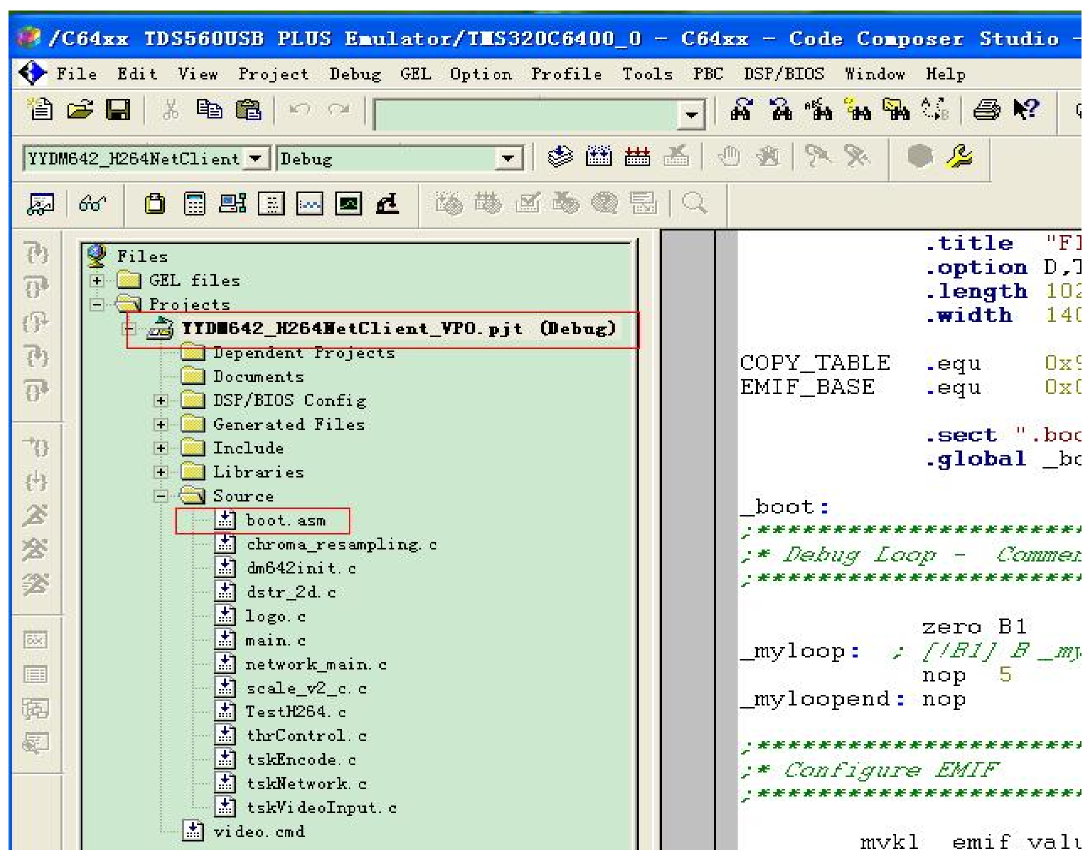
第三步，安装 Flashburn 工具。

在做完这些工作之后，我们现在学习用 CCS3.1 来烧写程序。

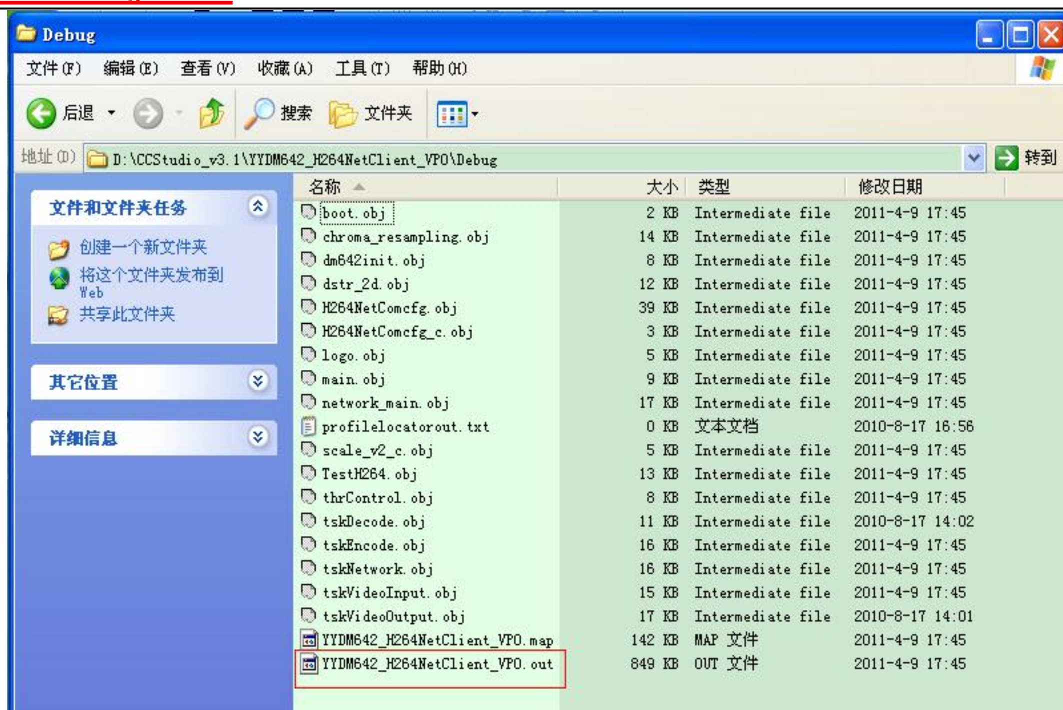
以一个网络视频传输程序的烧写为例来进行说明

第一步，把烧写需要的文件拷到当前工程目录中。主要包括 boot.asm、FBConfig_SP_31.cdd、FBTC642.out、hex6x.exe、LYDM642.cmd、LYDM642boot.bat 这六个文件。

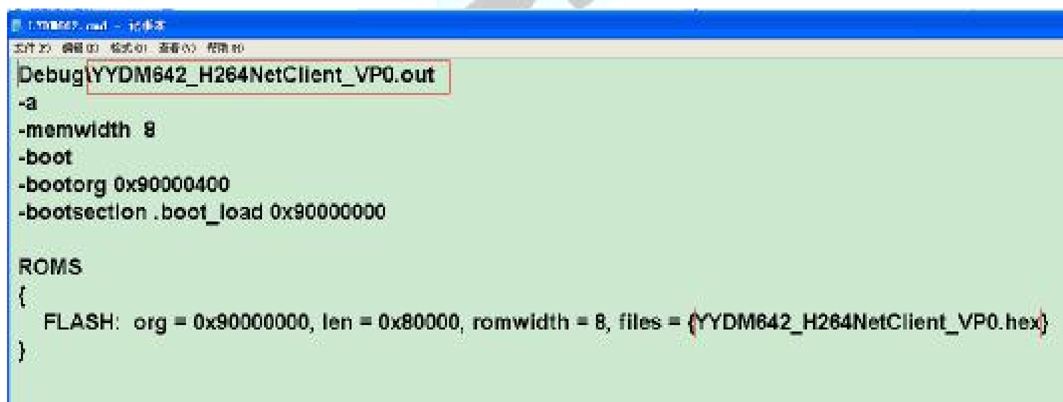
第二步，在 CCS3.1 中打开网络视频传输工程 YYDM642_H264NetClient_VP0，向工程中添加向量文件 boot.asm。如下图所示；



第三步，对该工程进行编译构建生成可执行的 out 文件。如下图所示；



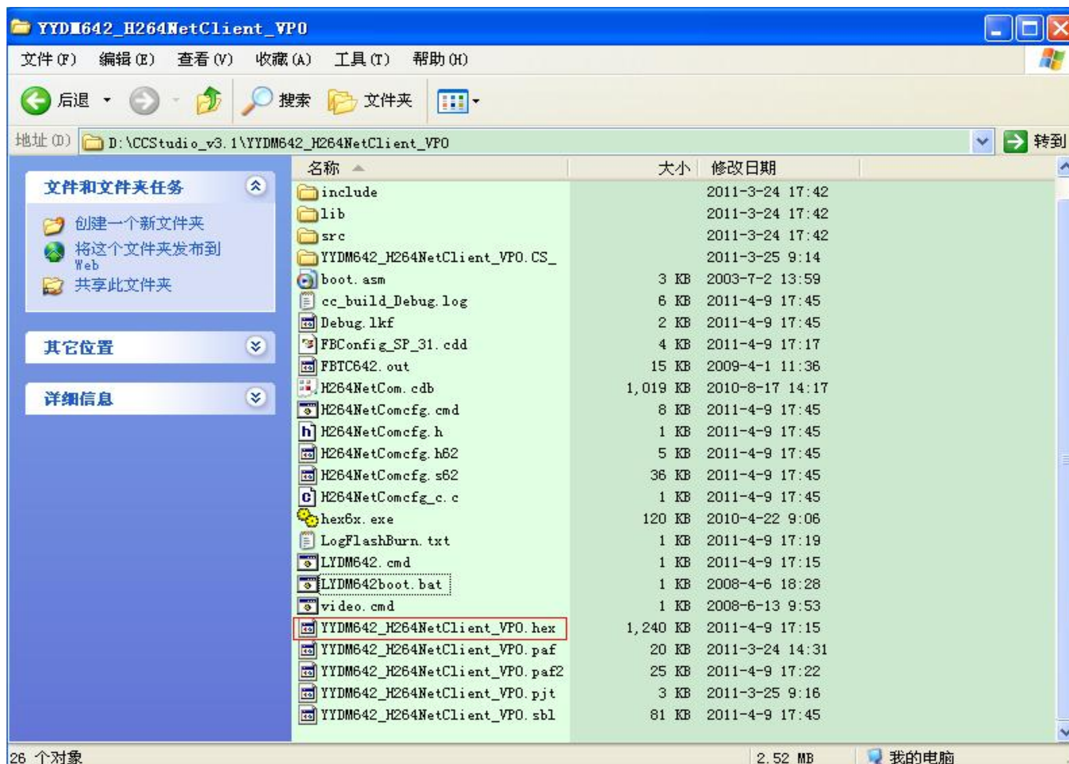
第四步，修改工程文件夹中的 cmd 文件，第一行改为编译该工程所生成的 out 文件名，
FLASH 这一行中改为希望生成的 hex 文件名。如下图所示；



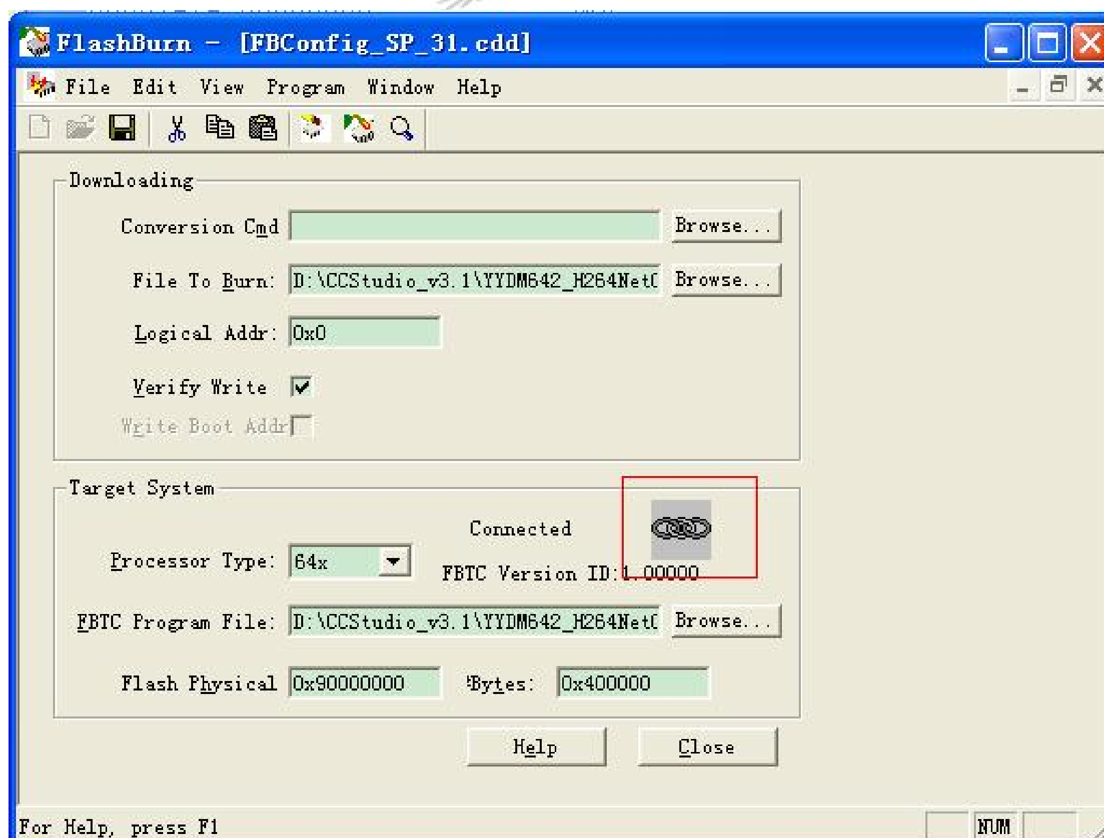
```
Debug\YYDM642_H264NetClient_VP0.out
-a
-memwidth 8
-boot
-bootorg 0x90000400
-bootsection .boot_load 0x90000000

ROMS
{
   FLASH: org = 0x90000000, len = 0x80000, romwidth = 8, files = {YYDM642_H264NetClient_VP0.hex}
}
```

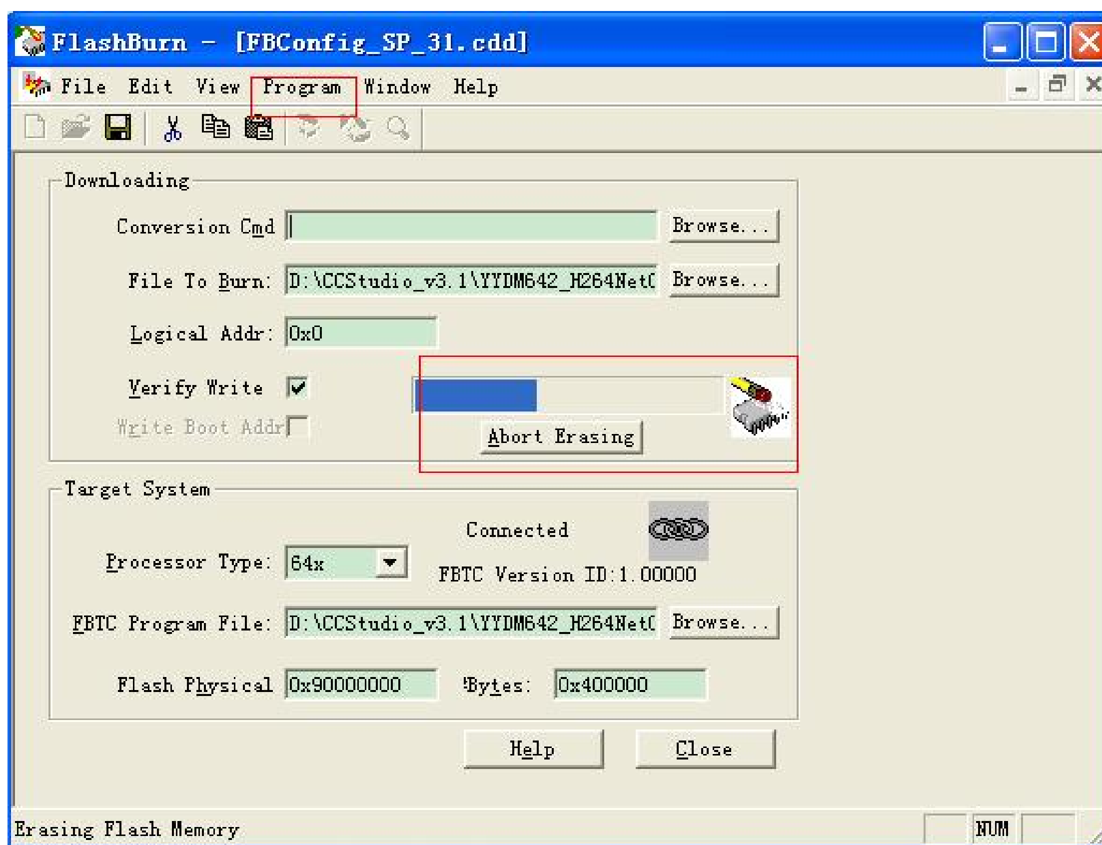
第五步，双击 LYDM642boot.bat，可以得到所需要的 hex 文件。



第六步，打开安装好的 Flashburn 工具，打开配置文件。注意连接，下图以红色标明



第七步，开始擦除 flash 中程序，选择 Program 菜单下的 Erase Flash。



第八步，向 flash 中写程序，选择 Program 菜单下的 Program Flash。
经过简单的几步操作，我们就可以把程序写进 flash 中了。

第四章　基于 YY-DM642 的图像和视频算法实现

4.1 图像阈值分割 Threshold

4.1.1　实验目的

1．　结合实例学习如何在视频显示程序中增加图像处理算法；

2．　理解和掌握图像的阈值分割原理；

4.1.2　实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台，DSP 仿真器

4.1.3　实验原理

灰度的阈值变换可以将一幅灰度图像转换成黑白二值图像。它的操作过程是先由用户指定一个阈值，如果图像中期权像素的灰度值小于该阈值，则将该像素的灰度值设置为 0，否则灰度值设置为 255。 灰度的阈值变换的变换函数表达式如下：

$$f(x)=\begin{cases}0(x<T)\\255(x\geq T)\end{cases}$$

其中 T 为指定的阈值。

4.1.4　实验步骤

1．连接好电脑和仿真器、YY-DM642 和仿真器；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；

3．打开 Threshold 文件夹里的工程文件 VideoProcess.pjt；

4．下载程序 out 文件，如果没有就先编译一下 Project－》Build （ ）；

5．打开 Videoprocess.c 文件，在如图 4-1 所示处设置断点；

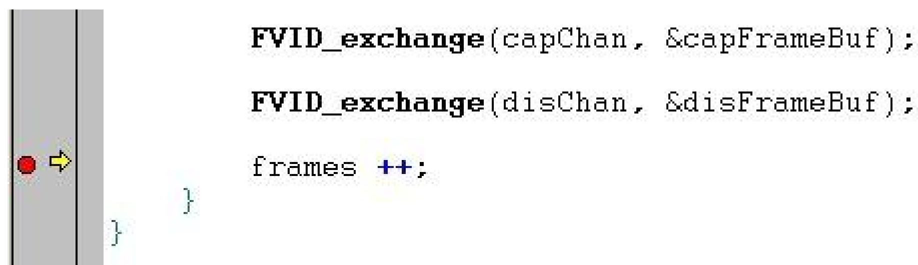


图 4-1　断点所在处

6．点击运行（ ），稍等片刻后程序停止在断点，通过 View－》Graph－》 Image

观察图像，对于本实验，按下图 4-2 所示设置 Graph Property Dialog ；

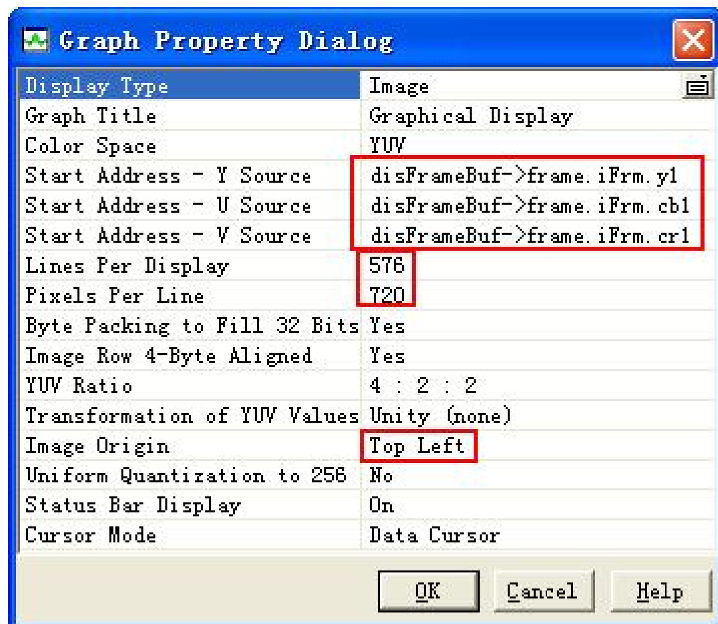


图 4-2 Graph Property Dialog 设置

7．本实验结束；

4.1.5 实验效果图



图 4-3 原图



图 4-4　灰度图



图 4-5　阈值分割图

4.2 灰度图的线性变换 LinerTrans

4.2.1 实验目的

1. 结合实例学习如何在视频显示程序中增加图像处理算法；
2. 理解和掌握图像的线性变换原理；

4.2.2 实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台，DSP 仿真器

4.2.3 实验原理

灰度的线性变换就是将图像中所有的点的灰度按照线性灰度变换函数进行变换。

该线性灰度变换函数 $f(x)$ 是一个一维线性函数：

$$f(x) = fA\ x + fB$$

灰度变换方程为：

$$D_B = f(D_A) = fA * D_A + fB$$

式中参数 fA 为纯属函数的斜率，fB 为纯属函数的在 y 轴的截距， D_A 表示输入图像的

灰度，D_B 表示输出图像的灰度。当 fA>1 时，输出图像的对比度将增大；当 fA<1 时，输出图像的对比度将减小；当 fA=1 且 fB ≠ 0 时，操作仅使所有像素的灰度值上移或下移，其效果是使整个图像更暗或更亮；如果 fA<0，暗区域将变亮，亮区域将变暗，点运算完成了图像求补运算。特殊情况下，当 fA=1，fB=0 时，输出图像和输入图像相同；当 fA=-1，fB=255 时，输出图像的灰度正好反转。

4.2.4 实验步骤

1．连接好电脑和仿真器、YY-DM642 和仿真器；
2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；
3．打开 LinerTrans 文件夹里的工程文件 VideoProcess.pjt；
4．下载程序 out 文件，如果没有就先编译一下 Project－》Build （ ）；
5．打开 Videoprocess.c 文件，在如图 4-6 所示处设置断点；

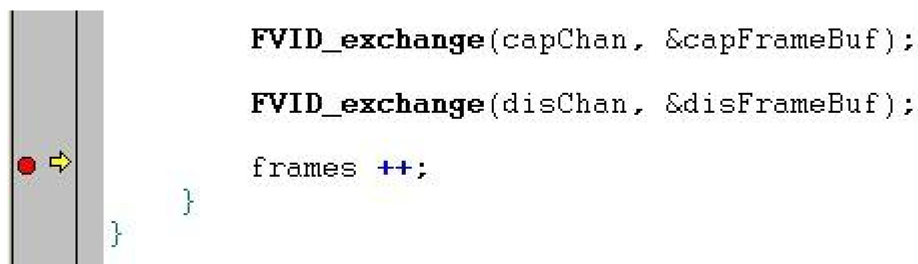
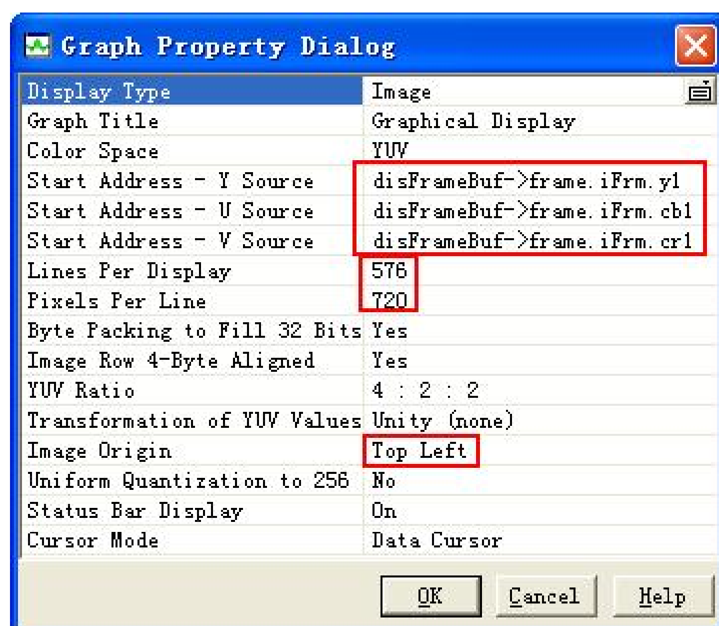


图 4-6 断点所在处

6.点击运行（ ），稍等片刻后程序停止在断点，通过 View—》Graph—》 Image 观察图像，对于本实验，按下图 4-7 所示设置 Graph Property Dialog ；



Graph Property Dialog

Property	Value
Display Type	Image
Graph Title	Graphical Display
Color Space	YUV
Start Address - Y Source	disFrameBuf->frame.iFrm.y1
Start Address - U Source	disFrameBuf->frame.iFrm.cb1
Start Address - V Source	disFrameBuf->frame.iFrm.cr1
Lines Per Display	576
Pixels Per Line	720
Byte Packing to Fill 32 Bits	Yes
Image Row 4-Byte Aligned	Yes
YUV Ratio	4 : 2 : 2
Transformation of YUV Values	Unity (none)
Image Origin	Top Left
Uniform Quantization to 256	No
Status Bar Display	On
Cursor Mode	Data Cursor

OK Cancel Help

图 4-7 Graph Property Dialog 设置

7. 本实验结束；

4.2.5 实验效果图



图 4-8　灰度图线性变换

4.3　灰度均衡 HistoEqualize

4.3.1　实验目的

1.　结合实例学习如何在视频显示程序中增加图像处理算法；

2.　理解和掌握图像的灰度均衡原理；

4.3.2　实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台，DSP 仿真器

4.3.3　实验原理

灰度均衡有时也称直方图均衡，目的是通过点运算使输入图像转换为在每一级上都有相同的像素点数的输出图像（即输出的直方图是平的）。这对于在进行图像比较或分割之前将图像转化为一致的格式是十分有益的。按照图像的概率密度函数（PDF，归一化到单位面积的直方图）的定义：

$$P(x)=\frac{1}{A_0}H(x)$$

其中 $H(x)$ 为直方图，A_0 为图像的面积。设转换前图像的概率密度函数为 $p_r(r)$，转换后图像的概率密度函数为 $p_s(s)$，转换函数为 $s=f(r)$。由概率论知

识，我们可以得到：

$$p_s(s) = p_r(r)\frac{dr}{ds}$$

这样，如果想使转换后图像的概率密度函数为 1（即直方图为平的），则必须满足：

$$p_r(r) = \frac{ds}{dr}$$

等式两边对 r 积分，可得：

$$s = f(r) = \int_0^r pr(u)du = \frac{1}{A_0}\int_0^r H(u)du$$

该转换公式被称为图像的累积分布函数（CDF）。

上面的分式是被归一化后推导出的，对于没有归一化的情况，只要乘以最大灰度值（D_{Max}，对于灰度图就 255）即可。灰度均衡的转换公式为：

$$D_B = f(D_A) = \frac{D_{Max}}{A_0}\int_0^{D_A} H(u)du$$

对于离散图像，转换公式为：

$$D_B = f(D_A) = \frac{D_{Max}}{A_0}\sum_{i=0}^{D_A} H_i$$

式中 H_i 为第 i 级灰度的像素个数。

4.3.4 实验步骤

1．连接好电脑和仿真器、YY-DM642 和仿真器；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；

3．打开 HistoEqualize 文件夹里的工程文件 VideoProcess.pjt；

4．下载程序 out 文件，如果没有就先编译一下 Project—》Build （）；

5．打开 Videoprocess.c 文件，在如图 4-9 所示处设置断点；

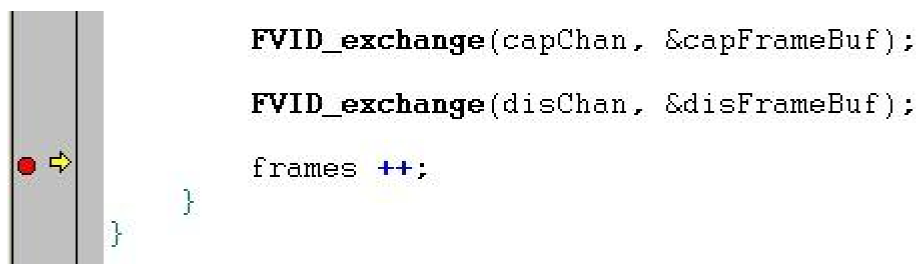


图 4-9 断点所在处

6.点击运行（ ），稍等片刻后程序停止在断点，通过 View－》Graph－》 Image 观察图像，对于本实验，按下图 4-10 所示设置 Graph Property Dialog ；

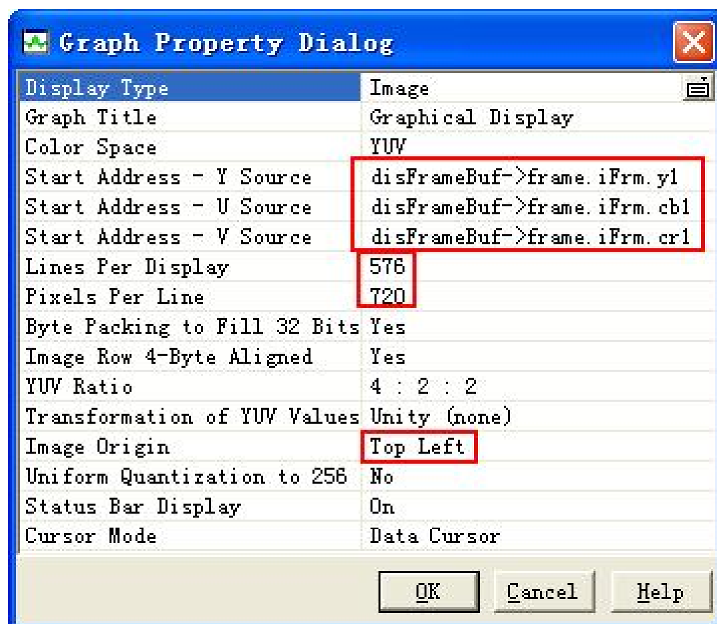


图 4-10 Graph Property Dialog 设置

7. 本实验结束；

4.3.5 实验效果图

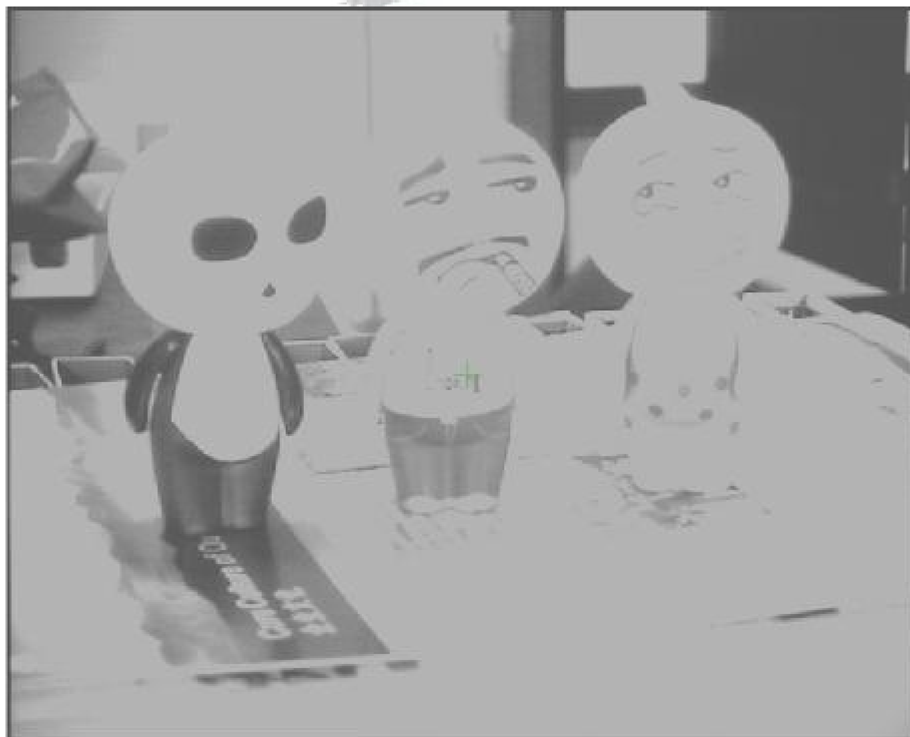


图 4-11 灰度均衡处理后的图像

4.4 图像的水平镜像变换 HorizTranspose

4.4.1 实验目的

1. 结合实例学习如何在视频显示程序中增加图像处理算法；
2. 理解和掌握图像的水平镜像变换原理；

4.4.2 实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台,DSP 仿真器

4.4.3 实验原理

设图像高度为 lHeight，宽度为 lWidth，原图中（x0，y0）经过水平镜像后变为（lWidth-x0，y0），其矩阵表达式为：

$$\begin{bmatrix} x1 \\ y1 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & lWidth \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x0 \\ y0 \\ 1 \end{bmatrix}$$

逆运算矩阵表达式为：

$$\begin{bmatrix} x0 \\ y0 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & lWidth \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x1 \\ y1 \\ 1 \end{bmatrix} \text{即} \begin{cases} x0 = lWidth - x1 \\ y0 = y1 \end{cases}$$

4.4.4 实验步骤

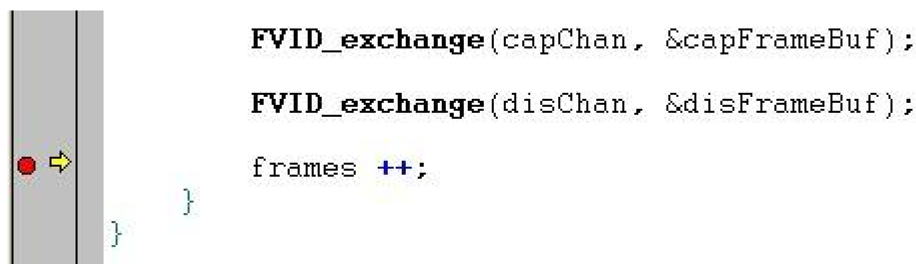
1．连接好电脑和仿真器、YY-DM642 和仿真器；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；

3．打开 HorizTranspose 文件夹里的工程文件 VideoProcess.pjt；

4．下载程序 out 文件，如果没有就先编译一下 Project－》Build （ ）；

5．打开 Videoprocess.c 文件,在如图 4-12 所示处设置断点；



```
FVID_exchange(capChan, &capFrameBuf);
FVID_exchange(disChan, &disFrameBuf);
frames ++;
    }
}
```

图 4-12 断点所在处

6.点击运行（ ），稍等片刻后程序停止在断点，通过 View－》Graph－》 Image

观察图像，对于本实验，按下图 4-13 所示设置 Graph Property Dialog ；

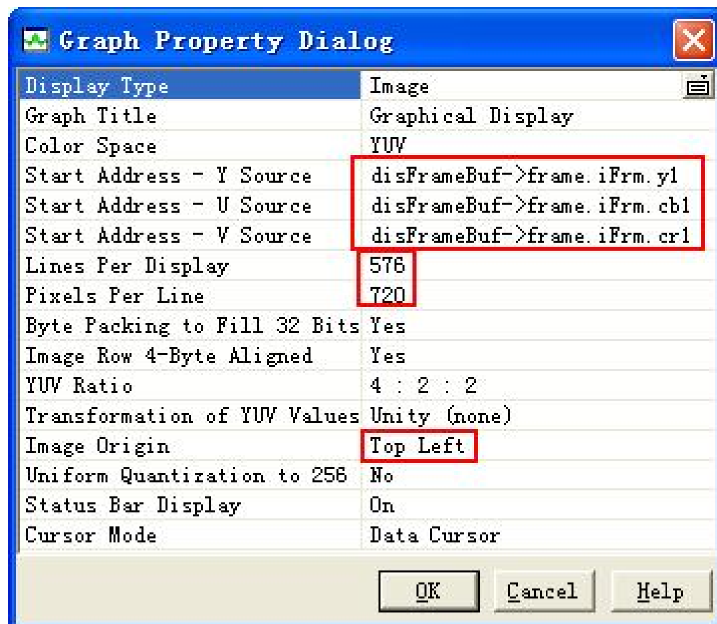


图 4-13 Graph Property Dialog 设置

7. 本实验结束；

4.4.5 实验效果图



图 4-14 图像的水平镜像结果

4.5 3*3 中值滤波 MedianFilter

4.5.1 实验目的

1． 结合实例学习如何在视频显示程序中增加图像处理算法；

2． 理解和掌握图像的中值滤波原理；

4.5.2　实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台,DSP 仿真器

4.5.3　实验原理

中值滤波是一种非线性的信号处理方法。中值滤波器在 1971 年由 J.w.Jukey 首先提出并应用在一维信号处理技术(时间序列分析)中，后来被二维图像信号处理技术所引用。中值滤波在一定的条件下可以克服线性滤波器如最小均方滤波、 均直滤波等带来的图像细节模糊，而且对滤除脉冲干扰及图像扫描噪声最为有效。由于在实际运算过程中不需要图像的统计特征，因此这也带来不少方便。但是对于一些细节多，特别是点、线、尖顶细节多的图像不宜采用中值滤波。

中值滤波一般采用一个含有奇数个点的滑动窗口， 将窗口中各点灰度值的中值来替代值定点(一般是窗口的中心点)的灰度值。对于奇数个元素，中值是指按大小排序后，中间的数值；对于偶数个元素，中值是指排序后中间两个元素灰度值的平均值。 对于一维情况，如图所示。它是用内含 5 个元素(1x5)的窗口对离散阶跃函数、斜坡函数、脉冲函数以及三角形函数进行中值滤波的示例。

4.5.4　实验步骤

1．连接好电脑和仿真器、YY-DM642 和仿真器；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；

3．打开 MedianFilter 文件夹里的工程文件 VideoProcess.pjt；

4．下载程序 out 文件，如果没有就先编译一下 Project－》Build （ ）；

5．打开 Videoprocess.c 文件,在如图 4-15 所示处设置断点；

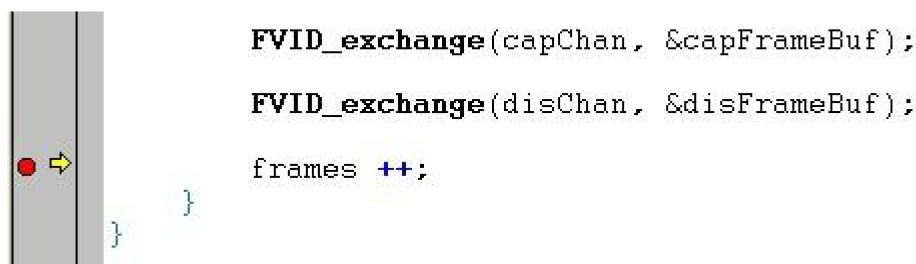


图 4-15 断点所在处

6.点击运行（ ），稍等片刻后程序停止在断点，通过 View—》Graph—》 Image 观察图像，对于本实验，按下图 4-16 所示设置 Graph Property Dialog ；

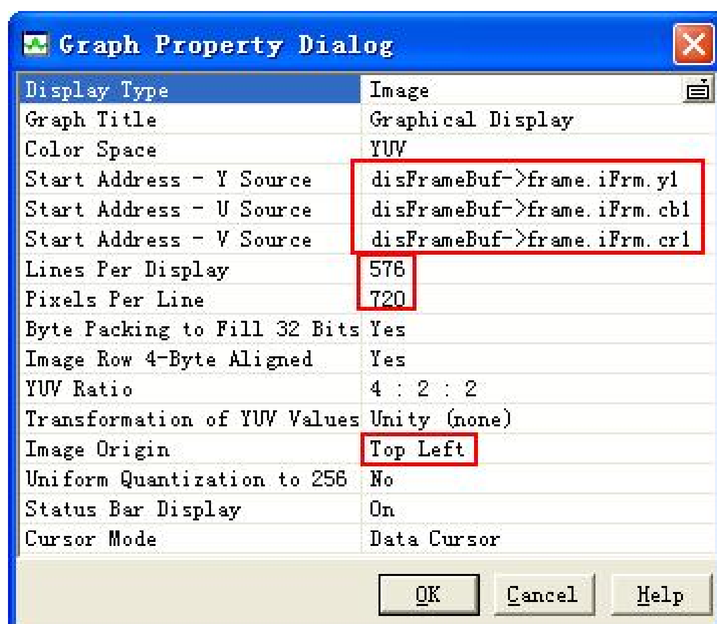


图 4-16 Graph Property Dialog 设置

7. 本实验结束；

4.5.5 实验效果图



图 4-17 3*3 中值滤波后的图像

4.6 边缘检测（Sobel 边缘算子）SobelEdge

4.6.1 实验目的

1. 结合实例学习如何在视频显示程序中增加图像处理算法；

2. 理解和掌握图像的 sobel 边缘检测原理；

4.6.2 实验设备

计算机，CCS3.1 版软件，YY-DM642 实验平台,DSP 仿真器

4.6.3 实验原理

利用计算机进行图象处理有两个目的：一是产生更适合人观察和识别的图象；二是希望能 由计算机自动识别和理解图象。无论为了哪种目的，图象处理中关键的一步就是对包含有大量各式各样景物信息的图象进行分解。分解的最终结果是图象被分解成一些具有某种特征的最小成分，成为图象的基元。相对于整幅图象来说，这种基元更容易被快速处理。

图象的特征指图象场中可用作标志的属性。 它可以分为图象的统计特征和图象的视觉特征两 类。图象的统计特征是指一些人为定义的特征，通过变换才能得到，如图象的直方图、频谱等等；
图象的视觉特征是指人的视觉可直接感受到的自然特征，如区域的亮度、纹理或轮廓等。利用这两类特征把图象分解成一系列有意义的目标或区域的过程称为图象的分割。

图象的边缘是图象的最基本特征。所谓边缘(或边沿)是指其周围象素灰度有阶跃变化或屋顶 变化的那些象素的集合。边缘广泛存在于物体与背景之间、物体与物体之间、基元与基元之间。
因此，它是图象分割所依赖的重要特征。

物体的边缘是由灰读不连续性所反映的。 经典的边缘提取方法是考察图象的每个象素在某个 邻域内灰度的变化，利用边缘临近一阶或二阶方向导数变化规律，用简单的方法检测边缘。这种方法称为边缘检测局部算子法。
边缘的种类可以分为两种：一种称为阶跃性边缘，它两边的象素的灰度值有着显著的不同； 另一种称为屋顶状边缘，它位于灰度值从增加到减少的变化转折点。下图分别给出了这两种边缘的示意图及相应的一阶方向导数、二阶方向导数的变化规律。对于阶跃性边缘，二阶方向导数在边缘处呈零交叉；而对于屋顶状边缘，二阶方向导数在边缘处取极值。

如果一个象素落在图象中某一个物体的边界上，那么它的邻域将成为一个灰

度级的变化带。

对于这种变化最有用的两个特征是灰度的变化率和方向，它们分别以梯度向量的幅度和方向来表示。

Sobel 边缘算子：下图所示的两个卷积核形成了 sobel 算子，图象中的每个点都用这两个核 做卷积，一个核对通常的垂直边缘相应最大，而另一个对水平边缘相应最大。两个卷积的最大值作为该点的输出位。运算结果是一幅边缘幅度图象。

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

4.6.4 实验步骤

1．连接好电脑和仿真器、YY-DM642 和仿真器；

2．给仿真器和 YY-DM642 上电（5V/2A），打开 CCS 并连接；

3．打开 SobelEdge 文件夹里的工程文件 VideoProcess.pjt；

4．下载程序 out 文件，如果没有就先编译一下 Project－》Build （）；

5．打开 Videoprocess.c 文件，在如图 4-18 所示处设置断点；

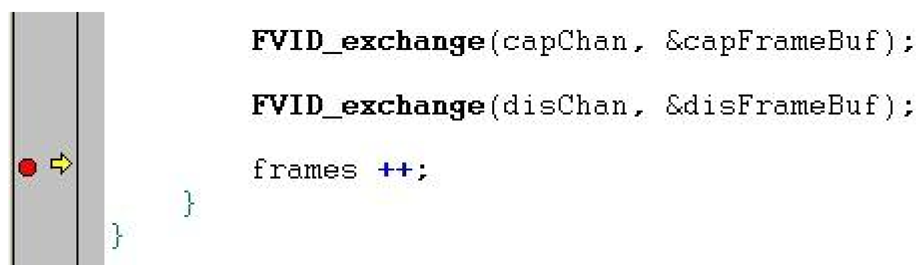


图 4-18 断点所在处

6．点击运行（），稍等片刻后程序停止在断点，通过 View－》Graph－》 Image 观察图像，对于本实验，按下图 4-19 所示设置 Graph Property Dialog ；

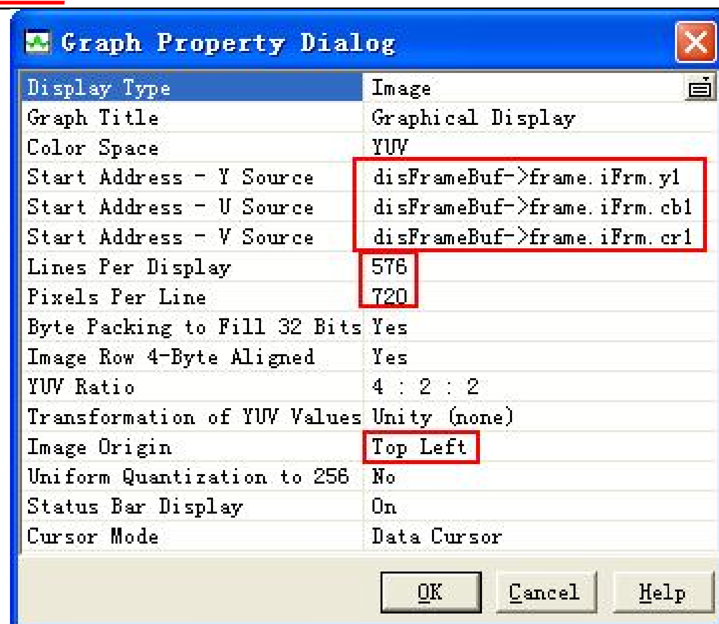


图 4-19　Graph　Property　Dialog 设置

7．本实验结束；

4.6.5　实验效果图

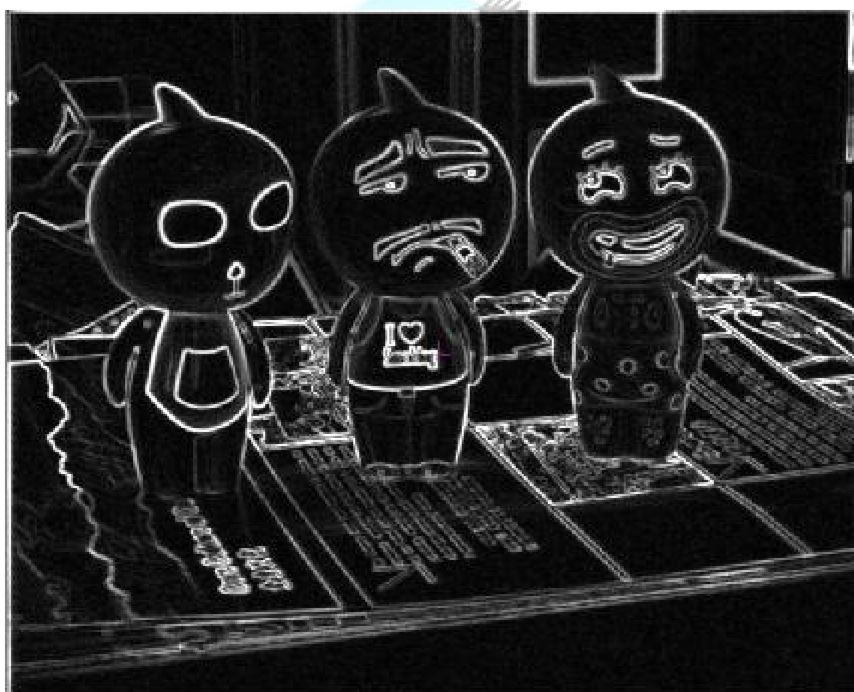


图 4-20　sobel 边缘检测后的图像

关于悦翼

悦翼长期从事嵌入式系统应用产品（实时操作系统、DSP/ARM 全系列开发调试工具、DSP/ARM 全系列开发板）、嵌入式图像跟踪识别系统及嵌入式技术项目的设计与定制等。嵌入式系统销售，产品涵盖 DSP，ARM，FPGA 等硬件平台。悦翼致力于将先进的图像工程理念和图像处理理论应用于具有特定要求或特殊应用场合的工程领域中，从工程实践的角度检验已有理论的有效性，并对已有理论做出工程化的评价、改进和创新。

在智能化视觉信息处理系统技术及应用集成方面，公司研发高性能嵌入式机器视觉应用开发平台，具有极强的处理性能，高度的灵活性，特别适用于工业检测、测量、识别、医学成像、网络视频监控、等高速 DSP 应用领域，在技术创新方面取得重大进展。在视频编解码领域有深入研究，对 MPEG4、H.264 视频压缩算法有稳定的研发队伍。

感谢您光临西安悦翼电子科技有限责任公司，如有任何需要了解和咨询的问题，请通过相应的服务热线联系我们，我们将会很快地跟您取得联系，同时为您提供最快捷、最方便、最优质的服务。

联系电话: 029-88839213，13991805685

联系人: 孙先生

网址：www.xauwing.com.cn

E-mail: sunweitom@tom.com/xauwing@126.com

联系地址: 西安雁塔区高新路 33 号新汇大厦 B2303D

邮编： 710000